Integration of Wireless Sensor Networks Into a Commercial Off-the-Shelf (COTS) Multimedia Network.

Molineux, Jeffrey S.

Monterey, California: Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

INTEGRATION OF WIRELESS SENSOR NETWORKS INTO A COMMERCIAL OFF-THE-SHELF (COTS) MULTIMEDIA NETWORK

by

Jeffrey S. Molineux

December 2008

Thesis Advisor:	Weilian Su
Second Reader:	John C. McEachen

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2008	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE Integration of Wireless Sensor Networks Into a Commercial Off-the-Shelf (COTS) Multimedia Network			**5. FUNDING NUMBERS**
6. AUTHOR(S) Jeffrey S. Molineux			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

As the primary military operating environment shifts from the traditional battlefields to a more diverse urban environment, the use of remote wireless sensors is increasing. Traditional development and procurement methods are not capable of meeting the changing requirements and time constraints of commanders. To minimize the time to develop and deploy new systems, commercial solutions must be examined. The focus of this thesis is on the integration of Commercial Off-the-Shelf (COTS) components into a wireless multimedia sensor network. Because components from multiple vendors were utilized, different operating systems and transmission protocols had to be integrated across the network. The network must be capable of providing a varying Quality of Service (QoS) level depending on the active sensors in the network. To ensure the QoS level is met, an adaptive QoS algorithm was implemented in the wireless IEEE 802.11 router which monitored and measured the outgoing transmission interface; from which, it determined the latency and transmission jitter. Based on the results, the program can adjust the bandwidth as necessary. Finally, a user interface is developed that allows end users to monitor the network. The performance of the network is based on the end-to-end throughput, latency and jitter exhibited by the network.

14. SUBJECT TERMS Wireless Sensor Networks, Sensors, Quality of Service, Adaptive Bandwidth			**15. NUMBER OF PAGES** 113
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

INTEGRATION OF WIRELESS SENSOR NETWORKS INTO A COMMERCIAL OFF-THE-SHELF (COTS) MULTIMEDIA NETWORK

Jeffrey S. Molineux
Lieutenant, United States Navy
B.S., Hawaii Pacific University, 2000

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2008

Author: Jeffrey S. Molineux

Approved by: Weilian Su
Thesis Advisor

John C. McEachen
Second Reader

Jeffrey Knorr
Chairman, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

As the primary military operating environment shifts to a more diverse urban environment, the use of remote wireless sensors is increasing. Traditional development and procurement methods are not capable of meeting the changing requirements and time constraints of commanders. To minimize the time to develop and deploy new systems, commercial solutions must be examined. The focus of this thesis is on the integration of Commercial off-the-shelf (COTS) components into a wireless multimedia sensor network. Because components from multiple vendors were utilized, different operating systems and transmission protocols had to be integrated across the network. The network must be capable of providing a varying Quality of Service (QoS) level depending on the active sensors in the network. To ensure the QoS level is met, an adaptive sampling algorithm was implemented in the sink node. The algorithm monitored and measured the incoming data packets from which it determined latency and transmission jitter. Based on the results, the program can adjust the sensor sampling rate as necessary. Finally, a user interface is developed to allow users to monitor the network. The performance of the network is based on the end-to-end throughput, latency and jitter exhibited by the network.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Remote Wireless Sensor Networks (WSN) are becoming of increasing interest to the military and intelligence communities. The ability to monitor a target area, without risking forces on the ground or deploying aircraft which may alert the target, provides a significant advantage to commanders. The majority of sensor platforms employed by the military were designed to monitor large targets such tanks and aircraft. These platforms lack the stealth and persistence required on the modern battlefield. Modern forces routinely operate in high density urban environments targeting small units or individual actors.

The ability to harness the capabilities of a large number of sensor platforms and distribute this information to forces on the ground is driving a considerable amount of research. In order to realize this concept of Network Centric Warfare, networks must be developed that are both robust in nature and easily deployed and operated. The traditional military procurements system can result in delays of years for badly needed sensor systems. Utilizing commercially available components would not only reduce the time it takes to develop and field a system but would decrease the cost per unit allowing more sensors to be purchased.

This thesis focuses on integrating commercial-off-the-shelf networking components into a WSN capable of meeting the demands of a military network. The network is broken down into four distinct segments. The end user devices provide output from the sensors in a graphical manner. The relay/bridge nodes serve as the long haul wireless backbone of the network. These components are responsible for relaying data collected in the field to forces in the field and headquarters elements. The sink node is responsible for aggregating the data collected by the sensors, translating the IEEE 802.15.4 datagrams into IEEE 802.11 datagrams and transmitting them to the relay node. The final segment is the sensor field. The sensors in the sensor field detect and measure the desired phenomena and relay the data to the sink node.

The percentage of packets lost in the end-to-end transmission of a data stream is of critical importance. If the percentage of packets dropped is too great, the validity of the measurements/detection may be called into question. To optimize the performance of a WSN network, designers must understand their target and be able to determine the exact requirements. Node density and sampling rate are critical parameters in a WSN. If a high node density is desired, the sampling rate must be decreased significantly. During experimentation, packet loss approaching 90% was experienced when both the node density and sampling rates were high. Effectively trading off between node density and sampling rate allows developers to maintain a packet loss percentage of less than 2%.

Because of the time sensitive nature of many targets, confirmation on the identity and location of a target of interest must be accomplished quickly. Streaming video provides one of the quickest means of verifying a target. Video applications are highly susceptible to excessive delay in a network. If the delay becomes too great, the buffer used for the arriving video packets may empty resulting in skipping or freezing of the video picture. Either situation could render the video signal unusable or of little value. As with packet loss, delay in the network is a function of both node density and sampling rate. Unlike packet loss, delay is more lenient in terms of node density versus sampling rate. Trials showed that at a given sampling rate networks of different node density tended to converge to one of three regions. Low density networks converged to a region around one second of delay. Networks with medium densities, between three and five nodes, converged to twelve seconds while higher density networks converged to fourteen seconds of delay. By determining the acceptable end-to-end delay, network designers can maximize the number of nodes they are able to deploy and still meet the requirements of the mission.

Voice Over IP (VOIP) is becoming a key feature on the modern battlefield. The ability to communicate with forces on the ground is critical in the modern operating environment. Shortfalls in traditional communications methods have caused a shift toward the use of VOIP. Integration of a VOIP capability into future WSNs is a desired feature. The network developed in this thesis is designed with VOIP in mind. Although not integrated into the current network, the strict tolerances demanded by VOIP are

considered when making design decisions. As with packet loss and delay, jitter is dependent on node density and sampling rate. Jitter seemed to converge to a minimum value of 100 ms when the sampling rate was held to 1 sample per second. Taking the nature of current targets into account, this sampling rate seems adequate to maintain continuity and fidelity in target tracking.

An Adaptive Sampling Algorithm is employed on the sink node in an attempt to shape the flow of packets across the network. This basic algorithm monitors the delay experienced between the sensor nodes and the sink node. Based on a threshold value, the sink node sends a message to the sensors telling them to increase or decrease their sampling rate. Using this algorithm, packet loss at the highest sampling rate was reduced from 60% to 7% in a network with four nodes. Delay was also reduced significantly. During initial network start up, delay began to exhibit the same upward trend as in earlier experiments. However, shortly after crossing the threshold level, the algorithm was able to bring the average delay back down to 100 ms. The end-to-end delay remained at this level for the remainder of the test period. The success of this algorithm proves the feasibility of using adaptive control schemes on individual nodes to shape the characteristics of the network.

In summary, the results obtained during the course of this thesis represent a first step in proving the concept of deploying a COTS mobile WSN in support of military operations.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

For my beloved wife Sonia, for her patience, understanding and sacrifice during the long hours of study, research and writing. Without your love and support I could not have finished this.

To my Thesis advisor Prof. Weilian Su, for his patience, guidance and friendship during this long process. Your probing questions and ideas helped to fuel my creativity and desire to advance this project. I look forward to continuing the collaboration on this project.

Thanks to Nate, Rob and Durke for serving as sounding boards for the last two years.

And to Olivia who endured many nights and weekends of not playing ball while I was occupied with my studies.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

The nature of modern warfare is dynamic and continually evolving. The traditional paradigm of peer-to-peer conflict between nation states is becoming less likely as globalization continues to blur international interest and boundaries. Failed states and stateless actors, such as Al-Qaeda, operating as small units in densely populated urban environments have replaced large standing armies operating in the open fields of Europe as the prevalent threat to national security.

As the nature of the threat has changed so have the primary targets of interest. The challenge for current and future battlefield commanders is not to locate and destroy large fixed targets such as tanks and aircraft, but to find, fix and destroy mobile, time sensitive targets such as individuals and small groups. It is these fleeting targets that will present the greatest challenge in future conflicts [1]. The majority of current military sensor platforms were designed to monitor and observe large conventional troop movements in broad areas. While space based and airborne sensors are effective at detecting vehicles moving across open areas, their capabilities are limited when it comes to detecting a lone individual moving across a city street. The time from when an event/target is first detected/located until the time when action is required has been greatly compressed. To help overcome the increased speed of battle, commanders must be able to leverage the most current information technology to develop and implement more dynamic planning procedures to synchronize forces across the battle space.

It is the nature of time sensitive targets, combined with the traditional stove piped methods of information collection and dissemination that can contribute to the greatest ambiguity and lack awareness in the battle space. As a result, military planners have begun to incorporate sensor networks into their operational and contingency planning. These networks can be quickly deployed in areas of interest to remotely monitor the movements and locations of friendly units as well as provide surveillance and intelligence

on hostile forces. Additionally, as the military finds itself increasingly tasked with Operations Other Than War (OOTW) sensor networks are being employed in disaster relief and humanitarian aid operations to provide situational awareness [2].

In this era of Network Centric Warfare (NCW), the ability to harness the power of large scale wireless sensor networks (WSNs) can allow near real time (NRT) dissemination of data across all levels of the battle space and can close the distance between decision makers and tactical units. To meet the varying needs of users at different levels; a scalable, reconfigurable, self-healing, high-performance information infrastructure must be developed to effectively link planners, decision makers and tactical forces [1]. Traditional wireless networking architectures, designed around the IEEE 802.11 family of standards, exist for many diverse situations and environments from corporate offices to coffee houses. However, the limited range of these traditional architectures makes them unsuitable for military WSNs.

The range of each node's radio dictates the maximum coverage a network may provide and the maximum distance the network can be from its wireless access point. The network must not only have the range to monitor the desired area of interest but must be able to relay the collected information to the end users. Due to the physical environment where military sensor networks are usually employed, these networks are generally wireless in nature. Because the wireless channel can pose significant challenges to reliable throughput, latency and jitter; military WSNs must address the Quality of Service (QoS) requirements in various environments and conditions.

Because the needs of planners at the theater level differ from those of the tactical unit preparing to enter a building, QoS protocols must be reconfigurable. Figure 1 provides a graphical depiction of the variable quality of service required across the command structure. At the strategic level, the number of users and the number of available data streams can be significant. However, due to the nature of the decisions made at that level, the timeliness and accuracy of the data does not need to be the same as at lower levels. Users can take the time to verify the accuracy of information before

making decisions. As the level of command moves closer to the tactical units, the number of available data streams decreases and the timeliness and accuracy of the data becomes more critical.

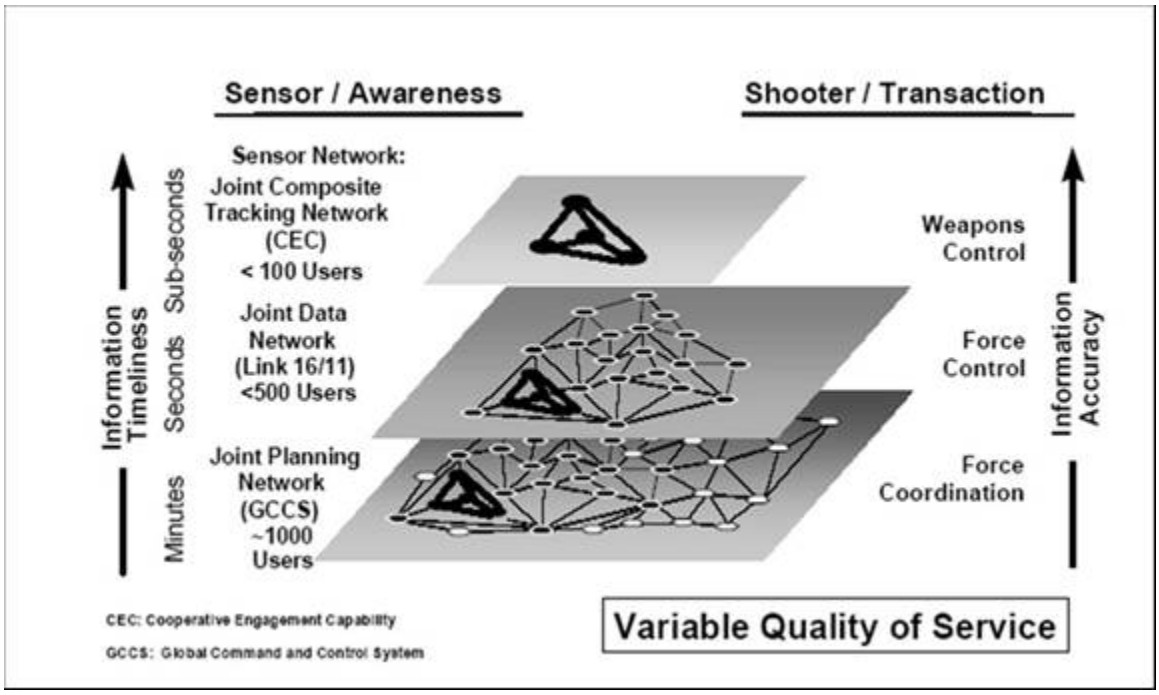

Figure 1. Variations in the Quality of Service Across the War fighting Enterprise. (From: 1)

Traditional networks generally employ point-to-point architectures that may have anywhere from one to a couple of hundred nodes. Military WSNs could employ hundreds to thousands of low-cost, low-powered sensor nodes. In this situation traditional network architectures would become overwhelmed. Using a multipoint-to-multipoint architecture would overcome the limitations of the traditional wireless architectures. Each sensor node would serve as a router as well as a collection point. Data streams are passed to the next router in the network where they are aggregated and further passed along until they reach the access point. By utilizing the sensor nodes themselves as routers, the power of the radio transmitters can be kept low, so the life of the battery can be extended. Also, the sensing range from the access point can be increased.

Since large scale WSNs employing point-to-multipoint architectures, such as mesh and ad-hoc networks, must aggregate large quantities of data, it is possible that critical data streams are dropped or corrupted. The best effort techniques employed by most commercial sensor networks do not meet the demands of military applications. To ensure critical data streams reach the final destination in a usable form, adaptive QoS techniques must be employed.

In the past, federal government organizations and agencies drove the research, development and application of new technologies. Today the commercial industry is the driving force behind the majority of ongoing Research & Development (R&D). To illustrate this point, Figure 2 shows a breakdown of R&D by sector for the fiscal years 1956 through 2006 using data provided by [3]. Until approximately 1980 the Federal Government led total percentage of R&D spending with a peak of 64% in 1966 compared with the commercial industry which only accounted for 33% of total spending. In the 1980's, this role was reversed with the commercial industry outspending the Federal government by a wider margin in each successive year. By 2006, the gap had widened to 37%, with commercial industry accounting for 65% of total R&D spending and the Federal government accounting for only 28% [3].

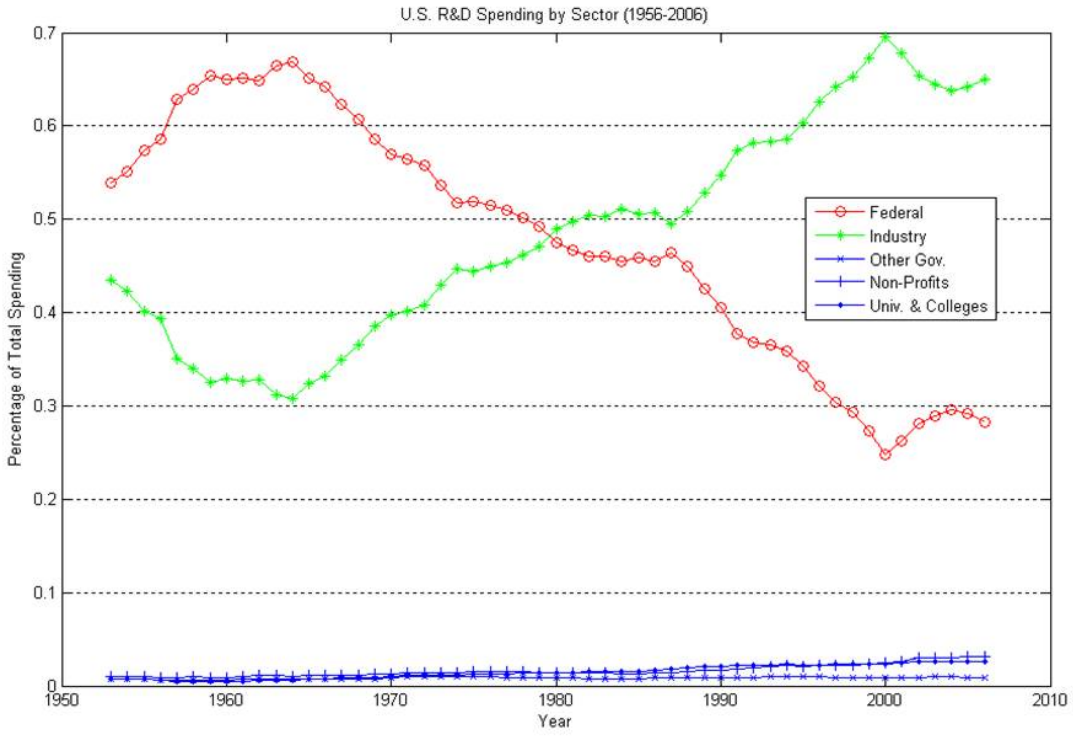

Figure 2. U.S. Research & Development Spending (By Sector) 1956-2006.

As a result of the substantial R&D investment by the commercial industry, many companies have developed networking equipment and solutions capable of meeting the needs of military planners. Revolutionary manufacturing techniques are producing equipment and sensors that are becoming increasingly smaller, more capable and cheaper to produce. It is becoming economically feasible to remotely monitor a wide array of phenomenon utilizing WSNs. Commercially produced sensor networks are currently being used in a wide array of applications from manufacturing to environmental to medical services [2]. These same sensors are easily concealable and adaptable to military missions. By utilizing a large grid of small wireless sensors the movements of individual actors can be tracked across a large urban environment.

Traditional military acquisition timelines are not capable of meeting the flexibility and demands of commanders. Adapting Commercial-off-the-Shelf (COTS) equipment for use in military applications has the added benefit of reducing the time it takes to develop and field new sensor networks. Since the equipment to be used in the network has already

been designed and constructed for use in civilian applications, military developers will only need to worry about modifying existing hardware and software. The time it takes to field new sensor networks could be reduced to months instead of years.

In addition to the extended military development and acquisition process, many new systems designed strictly for military applications involve the use of proprietary equipment and technology. This generally requires specialized training and additional maintenance, which increases the complexity and time it takes to field new technology. Additionally, the logistics support required could prove prohibitive in some operational environments.

By modifying COTS equipment for use in a military WSN, a reliable, scalable WSN can be developed and deployed for a fraction of the cost of acquiring a new military specific system. Damaged equipment can be easily replaced and reconfigured at locations worldwide. Widespread development of open source hardware and firmware allows for continual improvement and solutions to complex problems. Thus, it is possible to deploy WSNs that provide the QoS demanded across the war-fighting environment.

B. OBJECTIVES

The objective of this thesis is to validate the concept of adapting COTS multimedia devices into an integrated WSN, which is deployed to support military operations. The proposed concept will allow tactical users to connect to and monitor multiple sensor networks using small, commercially available, handheld devices and laptop computers. To extend the range of the network, a commercially available remote control helicopter with a mounted router will be employed, as shown in Figure 3, to serve as a relay node between the sensor field and the end users. The thesis concentrates on integrating multiple types of sensors into the network. Both IEEE 802.11 and IEEE 802.15.4 based sensor nodes will be utilized in the network. To allow the IEEE 802.15.4 sensor packets to be transmitted across the IEEE 802.11 segment of the network, a translator program was created for the sink node. User interfaces were created to allow interaction with the network. To test the performance of the network, various data streams were modeled to simulate expected network traffic.

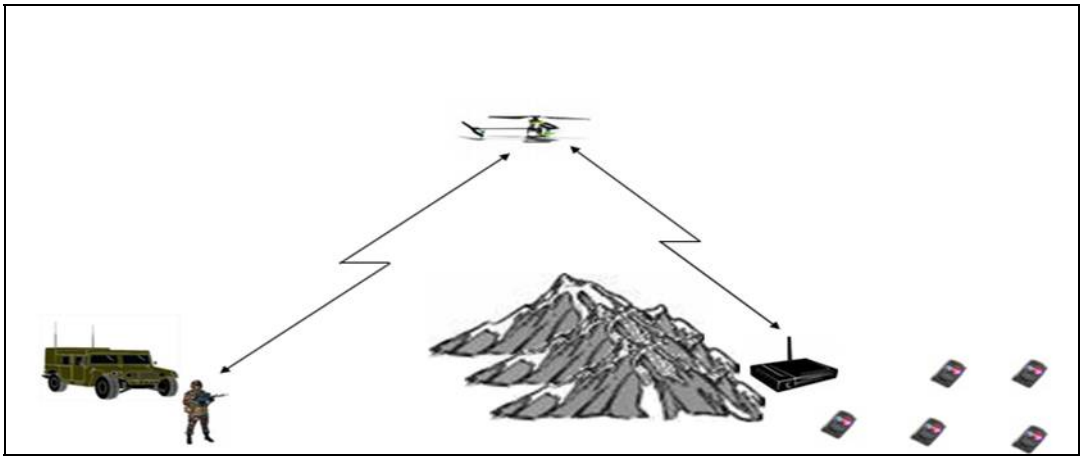

Figure 3. Integrated COTS Network Concept of Operations.

Wireless IEEE 802.11 routers were modified and reconfigured to allow for greater throughput and extended network range. Since the flow of data between the sensor node and the user is critical, adaptive QoS techniques and controls were developed and employed. These techniques involved active packet monitoring and an adaptive bandwidth adjustment protocol. The suitability of current QoS techniques was examined for use in the network. To the maximum extent possible, commercial applications were utilized to meet the needs and demands of the network and users.

C. THESIS ORGANIZATION

The thesis is divided into six chapters. Chapter I serves as an introduction and provides background and objectives of the thesis. In Chapter II, related research pertaining to the performance and integration of wireless sensor networks will be examined. Chapter III presents the prototype sensor network. Hardware and firmware/software choices and configurations are presented and examined. Adaptive bandwidth algorithms used to improve the performance and QoS of the network will be presented. The experiments and metrics used in the evaluation of the prototype networks performance will be discussed in Chapter IV. General models of the sensor traffic will be

presented along with the specific implementation. The results of these experiments will be presented and analyzed in Chapter V. Lastly, Chapter VI presents conclusions and recommendations for future work.

II. RELATED RESEARCH

Large scale WSNs have been the focus of much research in recent years. The ability to remotely monitor events with low cost sensors has led to prolific use in everything from military applications to home environmental monitoring [2]. Considerable time and money has been invested by the Federal government as well as commercial industry in improving the performance and lifespan of these networks while reducing the cost of production.

This chapter examines recent research pertaining to the development of WSNs. It begins with a look at basic research in the development and deployment of WSNs. Two deployments in support of environmental monitoring will be examined. The next section will look at improving performance through the design of the network architecture. Differing network topologies and routing algorithms are then explored in an effort to maximize throughput and increase the depth of coverage of the network. The chapter concludes with an examination of QoS techniques utilized in a wireless multimedia environment, to include adaptive bandwidth protocols.

A. WIRELESS SENSOR NETWORKS

Cross-discipline collaboration between engineers, computer scientist and domain scientist, such as biologist and environmental researchers, has led to the development of many application-driven WSN research projects. The ability to remotely monitor the behavior of animals and the environment with minimal human disturbance or interaction has provided both network and domain researchers with opportunities to gain a deeper understanding of the deployment and use of WSNs. WSNs have been used to monitor a wide range of environments, from active volcanoes in Ecuador [4] to bird habitats off the coast of Maine [5]. These deployments have led to significant gains in understanding the interaction between WSNs and the physical environment, something that could not be replicated in laboratory simulations.

In both deployments examined [4][5], researchers chose to use COTS equipment in order to minimize the time and cost of network development and deployment. Variants of the MicaZ sensor mote from the University of California at Berkley were used in both networks. The motes are easily configurable and operate on standard AA batteries. These deployments also demonstrated the wide range of sensors which could be utilized in a WSN. Application specific sensor boards were developed and integrated into the motes. Researchers off the coast of Maine chose to utilize a combination of sensors which operated in a discrete periodic manner. Each mote contained a combination of light, temperature, humidity and thermopile sensors. The sensors periodically transmitted discrete measurements which were relayed to researchers. By contrast, the researchers in Ecuador chose to utilize a combination of event driven sensors to record seismic and infrasonic signals. When a volcanic event was detected the sensors would record data in continuous time for a defined period of time. This data was buffered and relayed in near real time to researchers. While these two networks were limited to either discrete or continuous sensors, future WSNs could be configured with a combination of discrete and continuous sensors.

A significant difference between laboratory simulations and real world deployments was the relationship between the physical environment and the sensors. Researchers had to take into account with a wide variety of environmental factors when developing their networks. Sensors deployed to Great Duck Island, off the coast of Maine, were continuously exposed to rain, flooding, dew and low PH dense fog. In contrast, researchers in Ecuador were faced with a wet, humid, gritty, volcanic environment. In each case network engineers had to develop methods to protect the sensors from the environment while ensuring that the sensors could still perform and transmit as desired. Once again, researchers chose to utilize inexpensive COTS solutions such as parylene sealant and rugged cases vice expensive custom enclosures.

While the specific network topologies differed between the two projects, the basic concept was the same. Sensors would collect and transmit data to intermediate relay nodes which would then be used to transmit the data long distances to researchers. Researchers on Great Duck Island utilized a single-hop topology in the sensor fields.

Sensors placed inside the bird's burrows would collect and relay their data to gateway nodes just outside the burrow. These gateways would then transmit across a 350 foot link to a basestation node which was connected to the internet. Researchers at remote locations could then use the internet to access data from the sensor field. This four level architecture is shown in Figure 4.

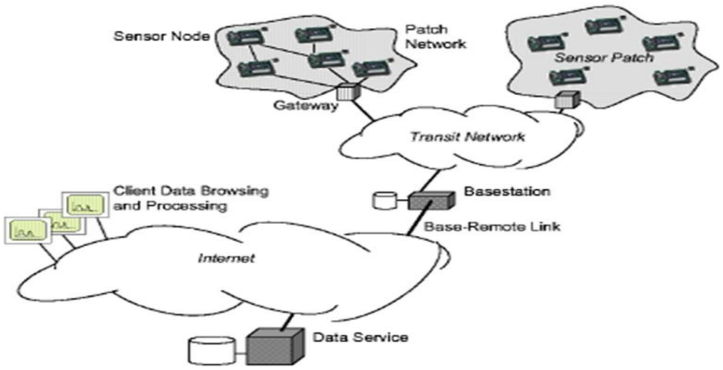

Figure 4. Great Duck Island Network Topology. (From: 5)

Network engineers in Ecuador chose a different architecture to meet their needs. Sensors were deployed in a linear topology on Volcán Reventador, as shown in Figure 5. Collected data would be relayed along the chain of sensor to a sink node were it would be aggregated. The aggregated data would be transmitted to a radio modem for long distance transmission to the researchers located approximately 4km from the sensor field.

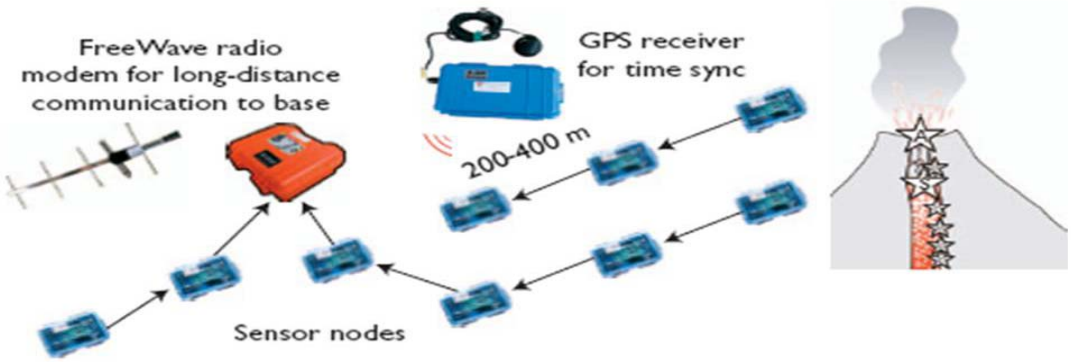

Figure 5. Volcán Reventador Network Topology. (From: 4)

Each of the network topologies examined exhibited its own unique set of challenges and problems. The linear topology and continuous time nature of the sensors used on Volcán Reventador presented a problem with excessive latency of the data. As an event would occur each node would begin collecting data. The end node would relay this data to the next node closer to the sink node. This node would aggregate the data and relay to the next node in the chain. As more data was aggregated, the limited buffering in the individual nodes, along with lack of time synchronization led to increased delay as the data passed through the chain. This resulted in high latency of data at the end user due to large throughput during each event. While this latency may not be a problem when data is collected for future analysis, in applications where real time warning is expected the chain topology could cause significant problems.

At Great Duck Island the one-hop topology prevented excessive latency due to buffering. However, researchers were faced with reduced end-to-end throughput in the network. Analysis showed that because each node transmitted to a gateway, which in turn transmitted to a sink node, the number of nodes within radio range of each other was significant. With the high density of network nodes in any given area the probability of packet collision was significant. As more nodes were added to the network the contention/collision problem increased. While the one-hop topology did not exhibit problems with latency the packet delivery ratio decreased as the number of nodes was

increased. These latency and packet delivery problems experienced during these two deployments demonstrate the need for a thorough understanding of the intended purpose of the WSN when designing a network topology.

Because the distance between researchers/users and the WSN can be significant, the ability to remotely monitor, diagnose and heal the network is critical. Properly designed user interfaces allow operators monitor the WSN and make any adjustments necessary. Simple Java based graphic user interfaces (GUIs) were developed to help researchers monitor the network on Volcán Reventador. The GUI allowed researchers to easily monitor the networks behavior and health, and manually set parameters such as sampling rate and detection thresholds to optimize the network. Well designed user interfaces allowed researchers working with the Great Duck Island project to predict node failures. Anomalous readings on the various sensors were easy to identify and were usually an indication of pending failure of the node.

B. NETWORK ARCHITECTURE AND ROUTING

As researchers in Ecuador and Maine discovered, selecting the proper network architecture and routing protocols is a critical aspect of network development. The chain topology used on Volcán Reventador allowed for greater depth of coverage but suffered from excessive latency during periods of high activity and provided very little breadth of coverage. Researchers on Great Duck Island were able to achieve good breadth of coverage using a one-hop topology but had to deploy more sensor clusters due to the limited depth of each cluster. The increased node density resulted in periods of excessive contention, which caused reduced throughput in the network.

In his thesis [6], Mak examined the end-to-end performance of the Star, Binary Tree and Chain network topologies when transmitting simulated video traffic. Utilizing Sun Microsystems Small Programmable Object Technology (Sun SPOT), video traffic was modeled using an on-off Pareto distribution. Network throughput, packet interarrival time, packet drop and packet delay were used to evaluate the performance of each topology.

The Star network topology, shown in Figure 6, allowed for good breadth of coverage but had limited depth. The network was configured as a one-hop network with the number of sensor nodes connected to the base station varied between one and six nodes. As would be expected, the end-to-end throughput of the network was directly related to the number of sensor nodes connected to the base station. Throughput decreased in an approximately linear manner from 1800 bytes/second to 211 bytes/second when the number of nodes increased from one to three nodes. Beyond three nodes the throughput remained relatively constant. Packet drop rate and packet interarrival times also converged as the number of nodes increased beyond four nodes. The decreased network performance can more than likely be attributed to the inherent characteristics of the base station. When the number packets waiting to be processed by the host application reaches 1000 the base stations firmware shuts the transceiver off [7]. The Star network topology would be suited for networks in which the sensor nodes sample periodically and transmit at a relatively low data rate such as long term environmental monitoring.

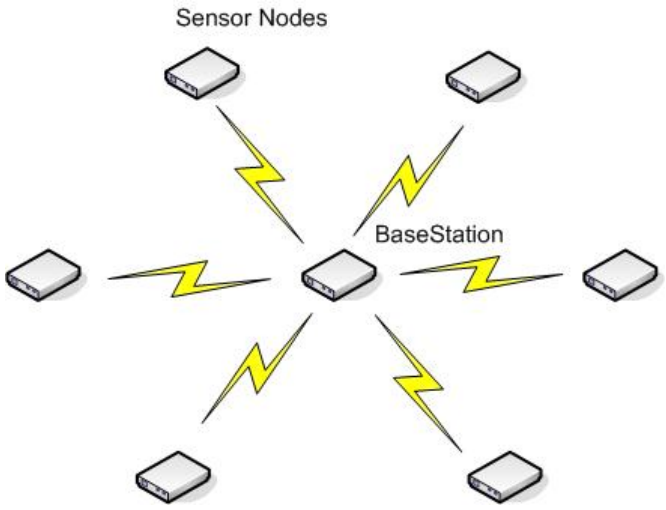

Figure 6. Star Network Topology

The greatest depth of coverage was achieved with the Chain network topology. In the Chain network topology, Figure 7, data is relayed from the outer sensor node to the next node closer to the sink node. Each successive node in the chain aggregates the

incoming data stream with its own data before relaying to the next node. Mak evaluated the performance of chains consisting of three and four sensor nodes. During the evaluation, the number of nodes was held constant while the shaping parameter, which affected the degree of traffic self-similarity, used in the modeling of the video stream was varied.

Intuitively, network throughput would be expected to decrease as the number of nodes is increased. As data streams are aggregated, the possibility of packet drop due to buffer overflow increases. The reduction in throughput was confirmed by experimentation, which showed the network's throughput decreased by approximately one-half when the number of nodes in the chain was increased from three to four. Additionally, the degree of traffic self-similarity also affected the network's throughput. Traffic patterns that exhibited lower degrees of self-similarity had the highest throughput. The combined effects of chain length and traffic self-similarity can have a significant impact of the networks throughput. The four node topology saw a reduction of two-thirds the throughput of the three node topology when the traffic pattern exhibited a high degree of self-similarity [6].

As the length of the chain is increased, the assumption would be that the end-to-end delay would be increased for all traffic patterns. Mak showed that the relative delay can not be assumed based on the length of the chain alone [6], the self-similarity characteristics of the traffic can cause unexpected behavior in the network. In the three-node chain topology, the higher the degree of self-similarity in the traffic, the longer the end-to-end delay. The four-node chain performs similar to the three-node chain when the traffic has a higher degree of self-similarity. As the degree of self-similarity decreases, the delay decreases until it reaches a threshold, when the shaping parameter was equal to 1.9, then the delay becomes greater than the self-similar traffic. Thus, when designing a network that is sensitive to delay, the type of traffic must be examined when determining the maximum depth of the network.

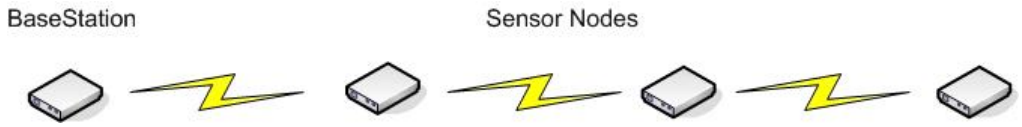

Figure 7. Chain Network Topology

The Binary Tree network topology, Figure 8, provides a good compromise between depth and breadth of coverage. Analysis of a two level, four-node network showed that traffic in the midrange of self-similarity, with Hurst parameter equal to 0.65 exhibited the best overall performance. The throughput of the Binary Tree topology is less than the Star topology but greater than the chain topology if four nodes are connected. Self-similarity of the traffic also had an impact on the throughput of the network. Traffic patterns on the extreme edges of self-similarity, such as Hurst parameters of 0.55 and 0.9, had the lowest throughput.

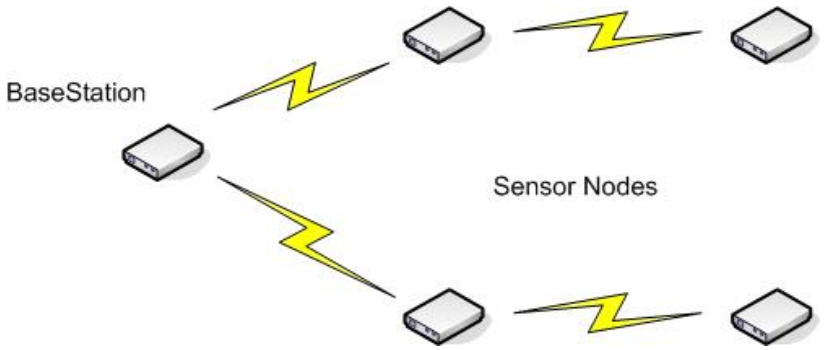

Figure 8. Binary Tree Network Topology

Delay in the Binary Tree topology was opposite that of the three-node chain topology. Traffic that exhibited a high degree of self-similarity had less delay than the four-node chain. The delay for traffic with a low degree of self-similarity was significantly longer. At the lowest degree of self-similarity, the delay was double that of the highest degree [6].

Routing protocol is another significant aspect of network design. Node mobility, density and network loading play a role in the selection of the proper routing protocol. Selection of the wrong routing protocol could result in poor network performance due to

dropped packets, excessive latency or under utilization of network resources. Routing protocols can be divided into two general categories, proactive and reactive. Proactive protocols attempt to maintain routes to each node in the network in a table. Reactive protocols determine routes only when they are needed. In [8], Pore examined three key ad-hoc network routing protocols, Destination-Sequenced Distance Vector (DSDV), Optimized Link State Routing (OLSR) and Ad-hoc On-Demand Distance Vector (AODV).

Mobility is a critical issue in modern military operations. Many targets operate in and around urban environments. The ability to monitor and track these targets of interest is crucial to the success of military operations. Packet drop is a key metric when operating in a mobile environment. The number of packets that are expected to drop increases as the speed of the mobile nodes increases. This is due to the increased number of timeouts resulting from link failure. In a mobile environment, the delay in reporting and propagating information about broken links results in dropped packets. Simulations in [8] showed the opposite, the Packet Delivery Ratio (PDR) for all three routing protocols increased as the speed of the node increased. The AODV protocol performed the best in the simulation with the PDR increasing from ninety percent at a speed of five meters/second to one hundred percent at a speed of fifteen meters/second. The PDR held relatively constant for speeds up to twenty-five meters/second. Both OLSR and DSDV had PDRs slightly under that of AODV but demonstrated the same performance pattern. As the node density was increased the PDR converged to between ninety-five and one hundred percent. The opposite was true as network loading increased. As the number of data connections was increased the PDR converged to around seventy percent with OLSR and AODV providing the best results. Increasing the speed of the connection had a more profound impact on the PDR. At a connection rate of fifty packets/second the PDR of all three protocols converged to fifty-five percent.

In time critical applications, end-to-end delay is of interest. In Ecuador [4], the delay experienced during periods of high activity prevented researchers from analyzing data until well after the event had concluded. In a military environment, this type of delay is unacceptable. As with the PDR, the delay rate for all three protocols improved as the

speed and density of nodes increased with a slight anomaly at a density of twenty nodes per area. The delay rate also worsened as the connection rate and number of connections in the network increased. The proactive protocols performed much better than did AODV.

C. QUALITY OF SERVICE

As the concept of Network Centric Warfare continues to be implemented on the battlefield, the number of data flows which are bandwidth sensitive is increasing. QoS sensitive applications, such as streaming video and VOIP, are common place in Iraq and Afghanistan. Deployed networks must be able to meet the high-speed, high-bandwidth demands of the new battle space.

QoS in networks has been the topic of considerable research over the years. Researchers developed the Integrated Services (INTSERV) and Differentiated Services (DIFSERV) protocols in an attempt to provide a measure of QoS in the Internet. These protocols face problems of scalability and efficiency, two issues inherently critical in military WSNs. New approaches to QoS, such as Adaptive Bandwidth Allocation have been the topic of much research in recent years. The ability to dynamically adjust the amount of bandwidth provided to a given data stream would allow for greater scalability and flexibility in deployed networks.

Two approaches to Adaptive Bandwidth Adjustment exist, node level and system or end-to-end level. In the node level approach, the metric of interest is measured at each node. Each node then implements a direct feedback algorithm [9] independent of each other. At the system level, the metric is measured at the end user device. The adjustment factor is then calculated and distributed throughout the network. Every node adjusts its bandwidth by the same amount at the same time.

In simulation [9][10], researchers evaluated a direct feedback algorithm to dynamically adjust bandwidth. Two different approaches were examined, node level adjustments and system level adjustments. The first series of test utilized a steady traffic rate. Both algorithms converged to the desired QoS level quickly, but the system level algorithm was able to more efficiently allocate the networks bandwidth. In the second

series of test, a dynamic traffic flow was applied to both algorithms. Once again, both algorithms converged to the desired level of QoS quickly and the system level algorithm provided the most efficient allocation of bandwidth. Comparing the results of both series of test shows that Adaptive Bandwidth Adjustment at the system level is the more efficient of the two methods. A more revealing method of examining the two approaches is to examine the number of QoS sensitive data streams each network was able to support. Table 1 shows the resulting number of voice sessions the network was able to support using each of the two approaches. The system level was able to support an average of six percent more voice sessions than the node level algorithm.

Simulation Runs	**Number of Voice Sessions Supported**	
	Node Level	***System Level***
1	244	259
2	259	267
3	239	269
4	253	269
5	248	255
6	249	268
7	255	267

Table 1. Voice Sessions Supported using Dynamic Bandwidth Allocation (After: 10)

D. SUMMARY

In this chapter, recent research in the development and employment of WSNs was examined, along with the performance of two different WSN deployments. Each demonstrated the need for careful consideration of the intended purpose when designing a WSN. Depth of coverage, latency and contention are critical metrics when developing the network. Recent thesis work covering network topology and routing protocols was

examined. Mak showed that the Binary Tree topology was the best compromise when both depth and breadth of coverage is desired in a network. Additionally, the degree of self-similarity of the networks traffic had a significant impact on the performance of the network. Depending on the topology and the specific metric, either a high or low degree of self-similarity resulted in decreased performance. The performance of three key routing protocols were also examined – AODV, DSDV and OLSR. The effects o mobility, density and network loading were examined for each of the protocols. AODV proved to have the best overall performance based on simulated results. The chapter concluded with an examination of Adaptive Bandwidth Allocation. Using a direct feedback algorithm, it was shown that a system level algorithm in which all nodes are adjusted by the same amount at the same time was the most efficient at allocating bandwidth.

Chapter III presents the network prototype. Components used in the development of the network will be examined to include hardware, firmware, and software. The chapter will conclude with an examination of the modifications and configuration changes incorporated in each component.

III. NETWORK PROTOTYPE

This chapter covers the design and configuration of the network prototype. The chapter begins with a detailed description of the components used in the prototype network and concludes with a discussion of the modifications and configurations implemented to improve performance and reliability.

The basic network architecture can be divided into four segments: user devices, relays/bridges, sink nodes and sensor nodes. User devices include any device an end user could user to monitor and interact with the network. Relays and bridges are used to extend the range of the network and deliver the transmitted signal to the user devices. Sink nodes collect, aggregate, and relay signals from the sensor nodes to the relay/ bridges. The final segment of the network is the sensor nodes that monitor and detect the desired phenomenon.

Figure 9 shows a graphical depiction of the network prototype and the data links beginning with the sensor nodes to the right and ending with the user devices on the left. As the sensors detect the desired phenomena they process data and transmit it through the sensor field using an IEEE 802.15.4 datagram. The data streams are aggregated at the sink node which then translates the packets from the IEEE 802.15.4 format to IEEE 802.11. The translated packets are then forwarded to the relays/bridges in the IEEE 802.11 segment of the network. The relay/bridge nodes apply adaptive QoS techniques, such as queuing disciplines and bandwidth adjustment, to ensure that packet delivery adheres to network performance requirements. The end user devices process the received packets and provide the result in a suitable form of output for the user.

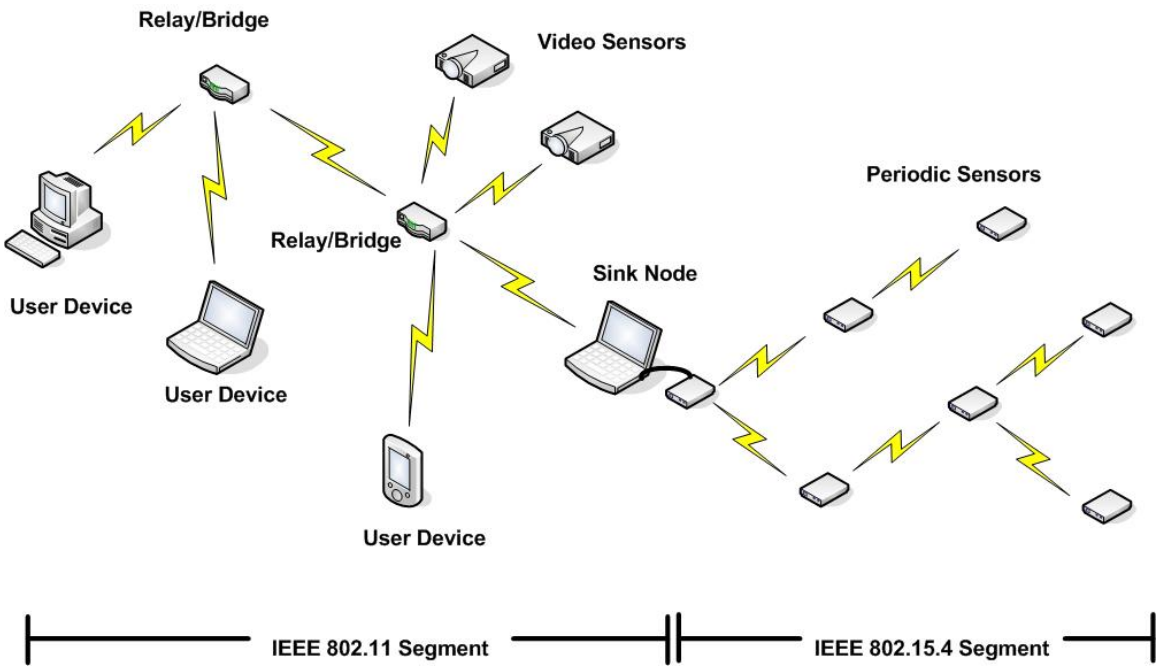

Figure 9. Network Prototype.

During the initial proof-of-concept phase the network will be restricted to three end user devices, a desktop computer, a laptop computer and a Portable Internet Tablet. Two relay/ bridges will be used to simulate the extended range provided by an airborne relay node. The initial network will consist of two sensor fields. The first will contain a single sink node and eleven periodic sensor nodes and will provide periodic data streams. The second field will provide streaming video. A simulated VOIP capability will be provided by two of the end user devices.

A. HARDWARE COMPONENTS

With the prevalence of wireless networks in the world, a wide variety of commercially available components exist from which to design and construct the network. Complexity and prices range from inexpensive generic units designed for wireless home computer networks to expensive reconfigurable units designed for corporate Wide Area Networks (WAN). To make the concept viable, the hardware

chosen had to be relatively inexpensive and commercially available worldwide. Additionally, the software/firmware had to be open source, to the maximum extent possible, to enable reconfiguration and further development.

1. User Devices

The User Device can be any device capable of transmitting and receiving an IEEE 802.11g wireless signal. The device allows users to connect to, configure and monitor the network and sensor data streams via a user interface panel. For the purpose of proof-of-concept, three separate user devices were used. For initial configuration testing, a Hewlett-Packard desktop computer operating Windows Vista was used to simulate a headquarters element using a wired connection to a router. The second user device was an Apple MacBook Pro laptop computer operating OS 10.5 (Leopard). The component tested the ability of mobile units to use an IEEE 802.11g capable laptop to interact with and monitor the network.

The final device integrated was a Nokia N800 Portable Internet Tablet. The N800 uses the 330 MHz Texas Instruments OMAP2420 Microprocessor [11]. The OMAP2420 contains a ARM1136 CPU core which utilizes a 32-bit RISC architecture [12]. The N800 provides 256MB of Flash Memory and 128MB of RAM for storage [11]. Two additional card slots are available to support SD memory cards up to 32GB apiece. The N800 is capable of wireless communication using either IEEE 802.11b/g protocols or the Bluetooth protocol. Wired communication is provided by a single USB 2.0 Mini-B connection which provides a limited supply of current (100mA). Power to the N800 is supplied by a rechargeable BP-5L Li-Po 1500 mAh Battery. User input to the N800 is accomplished using the built in 4.1 inch color touch screen.

The N800 uses the Linux based Maemo OS2007 operating system. The Maemo operating system uses a “light” version of Linux designed for mobile platforms with limited memory. The Maemo operating system does not contain all of the standard

libraries inherent in the full versions of Linux. While the N800 does support JAVA script, it is not JAVA compatible. These limitations restrict the amount of development possible on the N800.

2. Relay/ Bridging Nodes

The relay/ bridging node is responsible for relaying packets from the sink nodes to the end user devices using an IEEE 802.11g wireless signal. The Linksys WRT54GL V1.1 wireless router was chosen as the relay/ bridging node for the network. The WRT54GL is commercially available WI-FI capable gateway capable of supporting both the IEEE 802.3 Ethernet and IEEE 802.11 b/g wireless standards. The Linksys WRT54G family of routers is prevalent throughout the world, inexpensive and has significant third-party support in its development and modification. The GL model in particular was designed to be modified and reconfigured.

a. Hardware

The WRT54GL uses the 200 MHz Broadcom BCM5352 Microprocessor without Interlocked Pipeline Storage (MIPS) processor. The BCM5352 is a next-generation System-on-a-Chip (SoC) architecture that combines the CPU, Wireless MAC and Ethernet MAC onto one chip. The processor uses a Reduced Instruction Set Computer (RISC) architecture and supports data rates up to 125Mbps [13]. The unit provides 4MB of Flash Memory and 16MB of Double Data Rate Synchronous Dynamic Random Access Memory (DDR SDRAM) for storage [14].

For wireless communication, the wireless MAC of the BCM5352 is connected to the Broadcom BCM2050 2.4 GHz direct-conversion radio transceiver via the br0/eth1 internal interface. Figure 10 shows the internal block diagram of the WRT54GL. In addition to wireless communication, the WRT54GL provides five 10/100 Ethernet switches for wired communication. These ports are connected to the BCM5352 via the eth0 internal interface. By default all five ports are configured into two Virtual Local Area Networks (VLANs) in order to create multiple Layer 2 segments on the same Ethernet switch.

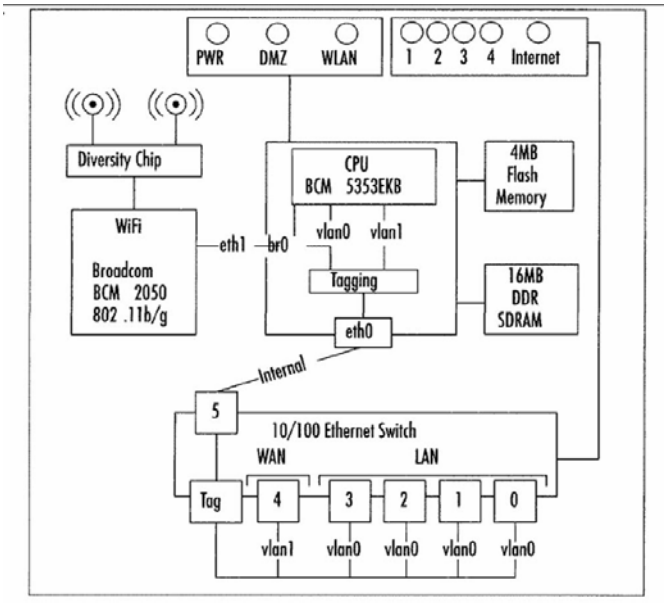

Figure 10. WRT54GL Internal Block Diagram. (From: 14)

Multipath distortion can cause significant problems in the wireless environment. Reflected copies of signals arriving at different times can either constructively or destructively interfere with each other. To improve the performance in a multipath environment, the WRT54GL comes standard with dual 2.2dB antennas connected to a diversity chip using Reverse Polarity-Threaded Niell-Concelman (RP-TNC) connectors. The diversity chip serves to improve the chance that the radio will receive a useable signal. Using two antennas separated by six inches, the diversity chip is able to sample each antenna and utilize the strongest signal. Figure 11 shows an example of how the dual antenna configuration helps to improve the performance of the WRT54GL in a multipath environment. As the drawing shows, reflected signals are received at both RX1 and RX2 in addition to the direct signal. Because of the difference in path length the phase shift of the reflected signals will differ at the two antennas. This difference in phase shift will cause different levels of degradation in the quality of the signal at each antenna. After sampling both antennas, the diversity chip would pass the stronger of the two signals to the radio.

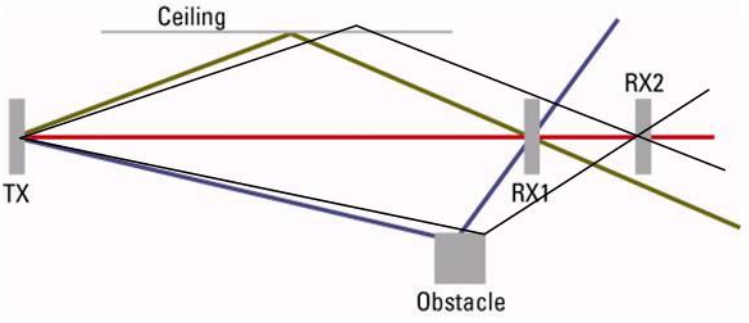

Figure 11. Dual Antennas Help Ensure That a Useable Signal is Received by at Least One Antenna. (After: 15)

Power is supplied to the unit by a 12V DC 1.0A wall adapter. Using a standard N-type connector and a short length of cable the power supply can be modified to operate on a wide variety of batteries from Lead Acid or Lithium-Ion batteries to standard AA alkaline batteries for operation in remote locations. While the Lead Acid battery last significantly longer than any of the other types of batteries as shown in Table 2, it is also the largest and most difficult to conceal [14]. Because the network is being designed for military operations, ease of concealment is important. Taking concealment into account, the Li-ion battery provides an adequate amount of operational time for the initial proof of concept with the benefit of a relatively small size.

Battery	**Approximate Rating**	**Approximate Length of Operation**
AA Alkaline (quantity 8)	2.2mAh (each)	4 hours
9V Alkaline	5mAh	2 hours
Li-Ion	2,200mAh	72 hours
Lead Acid	17.2 Ah	573 hours

Table 2. Battery Usage Times. (From: 14)

b. Firmware

The original Linksys firmware for the WRT54GL was created under the GNU General Public License (GNU GPL) which requires that the source code be made available to the general public. As of this writing the latest version of the Linksys firmware for the WRT54GL was version 4.21.1, updates and newer versions are available as free downloads from Linksys [16]. The firmware is Linux based and designed for use by the casual/novice home user and comes with a web accessible graphical user interface (GUI) for configuring the router.

Because of the popularity of the WRT54G series of routers numerous third-party firmware packages are available for free. Open WRT, DD-WRT, Fairuza WRT and Ewrt are just a few of the readily available third-party firmware packages. For the purpose of the network, two of the most popular firmware packages, Open WRT and DD-WRT, were compared with the original Linksys firmware. Both firmware packages are also Linux based and are available as free downloads [17][18].

Each of the three firmware package was evaluated based on a number of criteria from ease of use to suitability for advanced configuration and development. The two most important criteria were the ability of the firmware to support QoS and the suitability for advanced configuration. Figure 12 provides an overview of the elements evaluated and the availability of each element in the various firmware packages.

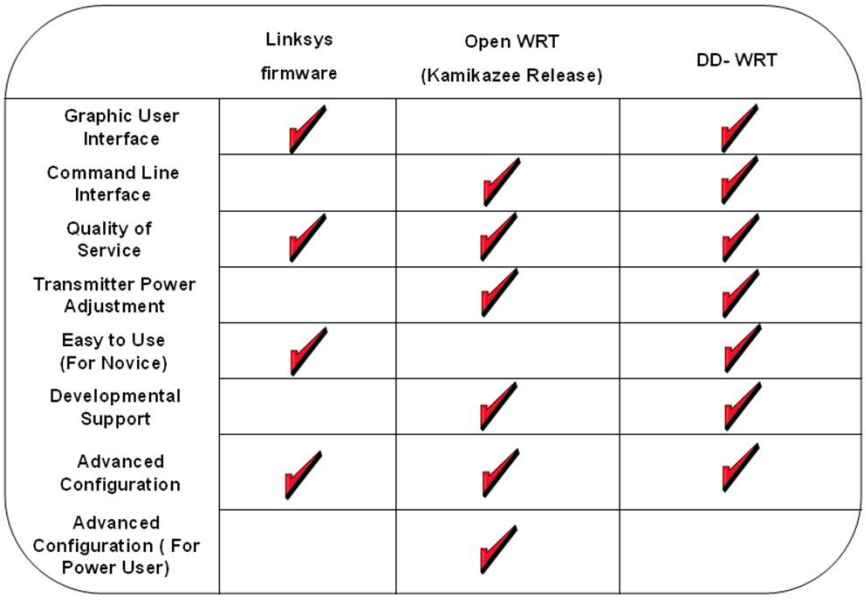

Figure 12. Linksys WRT54GL Firmware Comparison.

The first, and most obvious, element examined was the method of user control/configuration. Both the Linksys and DD-WRT firmware provided easy to use web accessible GUIs. The Open WRT package does not come standard with a GUI. Additional third-party software packages exist to provide a GUI but these proved troublesome to install and configure.

Both the Open WRT and DD-WRT come with command line interfaces (CLI) for advanced control and configuration. Because the Linksys firmware was designed for the novice user, a CLI was not provided with the standard firmware. While the CLI allows for advanced configuration of the router, many users in the field may not have the time or skills to utilize the CLI. Inclusion of both a GUI and CLI would enable technicians to provide initial advanced configuration of the unit while providing a means for the average user in the field to make any necessary adjustments.

The ability to support QoS features was a required feature of the firmware. All three firmware packages evaluated supported a variety of basic QoS techniques. Each of the firmware packages allowed delivery priority for an application to be set based on the type of application, incoming port number, incoming Ethernet port or the MAC

address of the sending node. Basic configurations allow for the user to specify high or low priority. For greater control, each package allowed the maximum bandwidth provided to an application to be set. Both the Open WRT and DD-WRT packages also allow for more advanced QoS configuration via the CLI. Using third-party Linux shell scripts or custom designed scripts, Linux libraries such as *tc* (transmission control) and *iptables* (Linux Firewall) can be used to fine tune the performance of the network. These libraries control the flow of packet traffic and establish filters, two capabilities that are critical in implementing adaptive QoS algorithms.

One of the most critical issues in a wireless sensor network is power consumption. Nodes in a sensor network must be able to operate on a limited supply of battery power. To maximize the lifespan, each node must minimize the amount of power consumed. One method of reducing some of the power consumed is to only use the minimum power necessary to successfully transmit to the next node. Of the firmware packages evaluated, only the DD-WRT package allowed the user to adjust the transmitter power. While optimizing power consumption is not an objective of this thesis, utilizing firmware with the ability to control the transmitters power will allow greater flexibility in future development of the network.

Although not a critical element, developmental support for the firmware package was desired to aid in future development. Both the Open WRT and DD-WRT packages enjoy significant support from the open source development communities. Websites and forums exist for developers to trade ideas and solve problems. This would allow network developers to gain access to solutions for firmware bugs and configuration problems allowing them to focus on developing more advanced configuration techniques. While the Linksys firmware was developed under the GNU, the company does not support additional development or modification to their firmware.

After comparing the available features and evaluating the ease of use and configuration, the DD-WRT firmware package is chosen for use in the relay/bridge nodes. The firmware is flexible enough to allow novice users to make adjustments easily while still providing the capability for advanced configuration and control. The DD-WRT

firmware is also the only package evaluated, which allows for the adjustment of the transmitter power, a feature desired in sensor networks that must operate on limited battery power.

3. Sink Node

The sink node serves as a gateway between the sensor nodes and the remainder of the network. The primary functions of the sink node are to aggregate and retransmit traffic it receives from the sensor nodes. In the prototype network, the sink node is also responsible for translating IEEE 802.15.4 packets into the IEEE 802.11g packet format. The Nokia N800 Internet Tablet was the first device chosen to be the wireless sink node for the network. Limitations with the Maemo operating system made the N800 unsuitable for the sink node. The lack of JAVA compatibility and limited Linux functionality prevented the development required to serve as a sink node.

A MacBook Pro laptop computer running the OS 10.5 operating system is used as an interim sink node to allow development of the remainder of the network. A SunSPOT basestation is connected to the laptop via a USB 2.0 interface to provide an input for the IEEE 802.15.4 packets while the installed Airport Extreme wireless card served as the IEEE 802.11g output interface. JAVA version 6 update 10 from Sun Microsystems and Another Neat Tool (ANT) version 1.7 from Apache are installed and serve as a foundation for the translation program [19][20].

4. Sensor Nodes

The sensor nodes are responsible for detecting and reporting the desired phenomenon. Each node contains the required sensors and an RF transceiver to relay collected data. For the proof-of-concept network, the Sun Microsystems Small Programmable Object Technology (SunSPOT) sensor motes are utilized. The SunSPOT is an experimental technology developed by Sun Microsystems to study and develop WSNs. SunSPOTS are sold commercially as development kits, which contain two free range (wireless) SPOTs, and one basestation SPOT that requires a USB 2.0 connection. The development kit also contains the SPOTManager software, used to manage and

configure the SunSPOTS, and the SOLARIUM Emulator for testing software. The free range SPOTs have the ability to be reprogrammed/reconfigured to perform a variety of tasks including periodic sampling and signal processing. Figure 13 shows the complete SunSPOT development kit [21].

Figure 13. SunSPOT Development Kit (From: 21)

a. Hardware

Each free-range SPOT comes with a processor board, which provides the basic operational functionality for the node, such as power control and radio reception/transmission. The SunSPOT uses the low-power, low-voltage Texas Instruments CC2420 single chip RF transceiver for communication. The CC2420 operates in the 2.4GHz spectrum utilizing the IEEE 802.15.4 protocol [22]. The sensitivity of the transceiver is -95dBm with a data rate of 250 kbps. The chip has two 128 byte FIFO buffers, one for transmission and one for reception. The chip uses direct sequence spread spectrum with a spreading gain of 9dB. Modulation is O-QPSK plus a half sine wave pulse shaping modulation and introduces a data latency of 2μs due to processing time [23].

Each SPOT can be uniquely identified by a 64-bit IEEE address similar to a conventional MAC address. These addresses begin with the Sun identifier 0014.4F01, followed by a 32-bit unique identifier. There are 255 ports available on each SPOT, although ports 1 through 31 are generally reserved for system functions. Communication between SPOTs uses standard connection oriented datagrams. Routing between SPOTs is accomplished via the reactive Ad-Hoc on-Demand Distance Vector (AODV) algorithm [24].

An eDemo sensor board which contains the sensors and input/output connections, shown in Figure 14, is also incorporated in the free range SPOTs. The eDemo board contains two switches, eight LEDs, a 3D accelerometer, temperature sensor, light sensor, six analog input terminals, five general purpose input/output pins and four high current input/output pins.

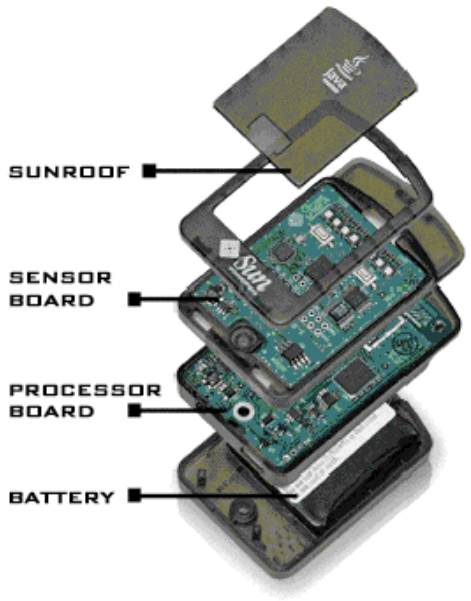

Figure 14. SunSPOT - Exploded View (From: 25)

The SunSPOT basestation serves as an interface between the free range SPOTs and the user device. The basestation uses the same processor board as the free range SPOTs but does not contain either the eDemo board or the battery. Power to the unit is supplied via the USB 2.0 connection.

b. Firmware

The SunSPOT uses a JAVA ME based operating system, which was developed under a GNU License. Code for the installed firmware is included with the Software Development Kit (SDK) or can be downloaded from [21]. Firmware can be upgraded/updated using the either the SPOTManager software included with the development kit or via a CLI using ANT commands. As of this writing, the current stable release version of the SDK was v4.0 "blue". A development version of the SDK, "red", is available and is utilized in the development of the network.

The SunSPOT firmware uses the "Squawk" Virtual Machine (VM), which extends the JAVA ME MIDlet class, to execute the JAVA byte code. Unlike the VM's used in other JAVA mobile devices, the Squawk is capable of running multiple applications at the same time by separating them into *Isolates*. Each *Isolate* can contain multiple JAVA threads and objects. This allows each SPOT the flexibility of running multiple applications at the same time.

Extensive support for the development and modification firmware is available in the forums and blogs operated by Sun Microsystems and numerous third-party developers [25] [26] [27]. Sun Microsystem's Project Sun Spot development team regularly monitors the forums and provides support and solutions for system bugs and general development problems.

c. Software

Software applications for the SunSPOTs are divided into two segments, the host segment and the SPOT segment. The host segment resides on the computer that will be used to monitor the output of the application. This segment is written in standard JAVA SE. The SPOT segment is the application code that is deployed on the SPOT. This code is written in JAVA ME and uses the MIDlet lifecycle.

Software applications for the SunSPOT can be developed in any Integrated Development Environment (IDE) which supports JAVA ME or in a basic text editor. The simplest IDE to use is the Netbeans IDE, which is supported by Sun

Microsystems. Plug-in applications are available, which provide development templates for both the host and SPOT segments of the application. As of this writing, the current release is version 6.5; however, this version is not preferred for developing SPOT applications as it has a software bug that prevents reliable compiling of JAVA ME programs. The older version 6.1 is stable and preferred for developing JAVA ME applications. Current and older versions of the IDE can be downloaded at [28].

5. Video Sensor Nodes

The video sensor nodes are stand-alone devices which provide streaming video to remote users. The D-Link DCS-5220 Pan/Tilt Internet Camera was chosen for the proof-of-concept network. DCS-5220 provides a live video feed at 30 frames per second over either IP based or 3G based networks. The camera provides a 270 degree horizontal and 90 degree vertical viewing range [29]. The optical sensor has a 0.5 Lux sensitivity and a 4X digital zoom. Output can be provided using either a wired IEEE 802.3 Ethernet connection or a wireless IEEE 802.11b/g connection. Output can be provided in either a MPEG4 format for streaming video or a JPEG format for snapshots.

B. COMPONENT MODIFICATION AND CONFIGURATION

Military WSNs must be capable of providing a reliable QoS level to the end users while maintaining ease of use. The COTS components used in the network were not capable of meeting the performance demand of a military WSN in stock configuration. To meet the requirements of the concept network, modifications and configuration changes were made to the firmware and software of the components.

1. User Devices - Desktop/Laptop User Interface

To monitor and interact with the network a GUI was created for use on desktop and laptop computers. Written in JAVA the GUI provides a basic framework for the current and future versions of the network. Figure 15 shows the main user interface panel. The panel is broken into three basic areas based on functionality. On the right side of the panel is a large viewing area for streaming video. The lower left section toggle buttons

and sliders for audio sensors and VOIP applications. At this time functionality of the video, audio and VOIP sections is limited and the components serve as a framework for future development.

The upper left half of the panel provides monitoring for the periodic sampling data streams, in this case the temperature and light intensity readings. These readings are displayed in the boxes located below the toggle buttons. Each of these toggle buttons opens an additional user interface, Figure 16, which allows users to monitor the light and temperature sensors in greater detail and in real time monitoring. Values are plotted in the upper panel of Figure 16 to provide a graphical representation of the readings while the lower panel presents the readings in text form. Due to limitations in JAVA, the current light/temperature display interfaces are limited to eight individual sensors. For each sensor above the eight, the results are combined in the eighth panel.

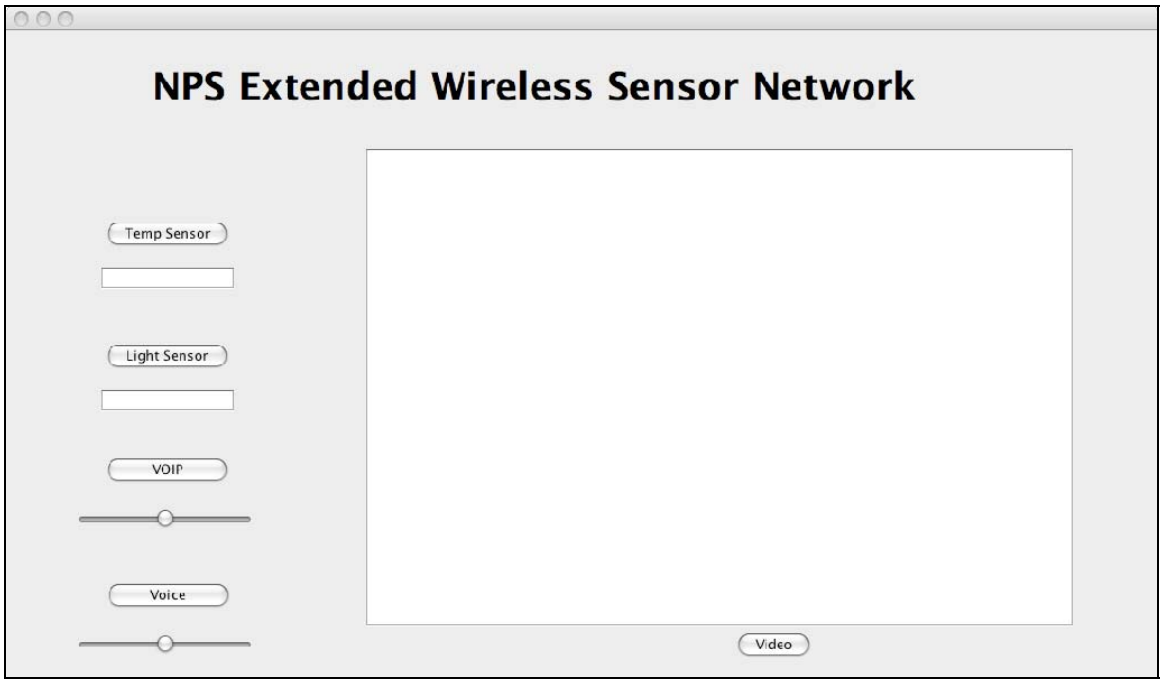

Figure 15. Desktop/Laptop User Interface

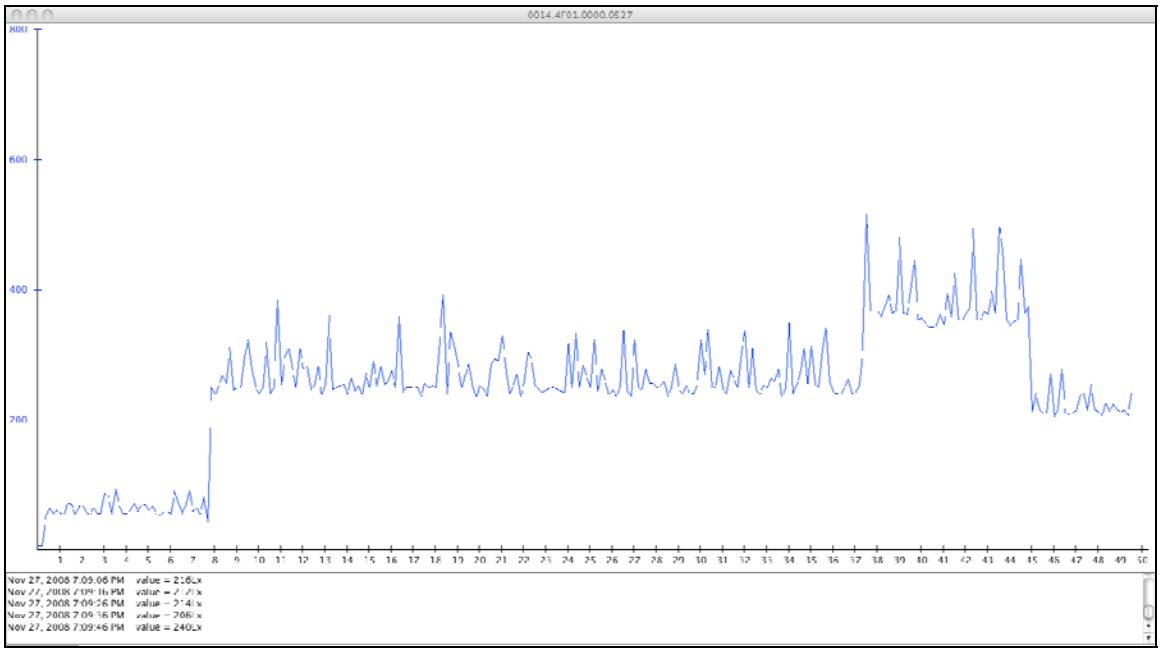

Figure 16. Desktop/Laptop Periodic Sampling User Interface

2. Relay/Bridging Nodes

a. Firmware Update

Firmware on both of the WRT54GL routers was upgraded to meet the performance requirements of the network. DD-WRT firmware package versions V23, V24 RC6, V24 RC7 and V24 SP1 were evaluated for performance and reliability. While the V24 SP1 package is considered the current stable release, operation in the wireless repeater mode is intermittent, requiring the router to be rebooted regularly. The V24 RC6 package is unstable when operated on the WRT54G family of routers. The firmware was evaluated on two WRT54GL V1.1 routers and one WRT54G V8 router. In each case the firmware caused the router to "brick", a condition in which the router suffers a failure which requires the firmware to be reloaded to operate the router. While considered stable, the V23 is obsolete and does not fully support the wireless repeater mode. The only package to support all network requirements and proved stable was V24 RC7.

Upgrading the firmware for the WRT54GL can be accomplished using either the web based GUI or CLI. If the router is configured with the standard Linksys firmware, special procedures must be used to install third-party firmware. The standard Linksys firmware limits file downloads to a maximum of 3MB in size. Because all of the firmware versions examined, with the exception of Linksys variants, were greater than 4MB in size, they can not be directly downloaded into the router. To get around this limitation, the DD-WRT micro version firmware can be installed first. Once the micro version is installed, the limitation of download file size is removed and the full firmware version can be installed. Detailed instructions on installing the DD-WRT firmware package can be found at the DD-WRT website [30].

b. Wireless Bridge/Repeater

The concept of operations for the network incorporates a remote control helicopter to extend the range of the network. Signals transmitted by the relay node on the ground are passed to a relay node mounted to the bottom of the helicopter. This node serves as an airborne wireless repeater to extend the range of the network beyond the limited range of the low power transmitters in the WRT54GL. The DD-WRT V24 RC7 firmware package supports both wireless bridging and wireless repeater bridging as a standard feature.

Wireless bridging utilizes both wired and wireless segments. The primary router can communicate with clients either by wired or wireless IEEE 802.11. The primary router relays packets from these clients to a secondary router using an IEEE 802.11 connection. The secondary router can only relay received packets to clients connected via an IEEE 802.3 Ethernet connection. Figure 17 shows a graphical depiction of a wireless bridge with the primary router is on the left, while the secondary router is on the right [31].

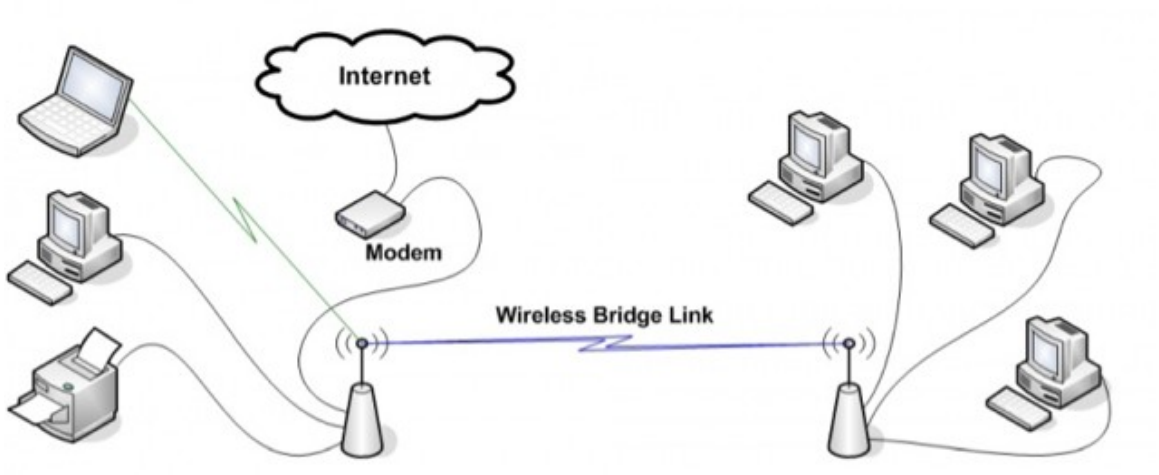

Figure 17. Wireless Bridge (From: 31)

The wireless bridge repeater is similar to the wireless bridge, but allows clients to connect wirelessly to both the primary and secondary routers using an IEEE 802.11 transmitter. Figure 18 shows a graphical depiction of the wireless bridge repeater. The only difference between the wireless bridge and the wireless repeater is the ability of the secondary routers clients to connect wirelessly. Using the router as a wireless bridge repeater does reduce overall throughput of the router by approximately fifty percent to clients connected to the primary router. This is due to the fact that the transceiver in the primary router is used to wirelessly receive and relay packets between two nodes [32]. At this stage of integration the reduced throughput does not pose a significant problem for the network. The scale of the network is small enough that the low data rate sensors do not overwhelm the networks capacity. As the scale of the network is increased, a maximum number of sensors that can utilize a relay node will need to be determined.

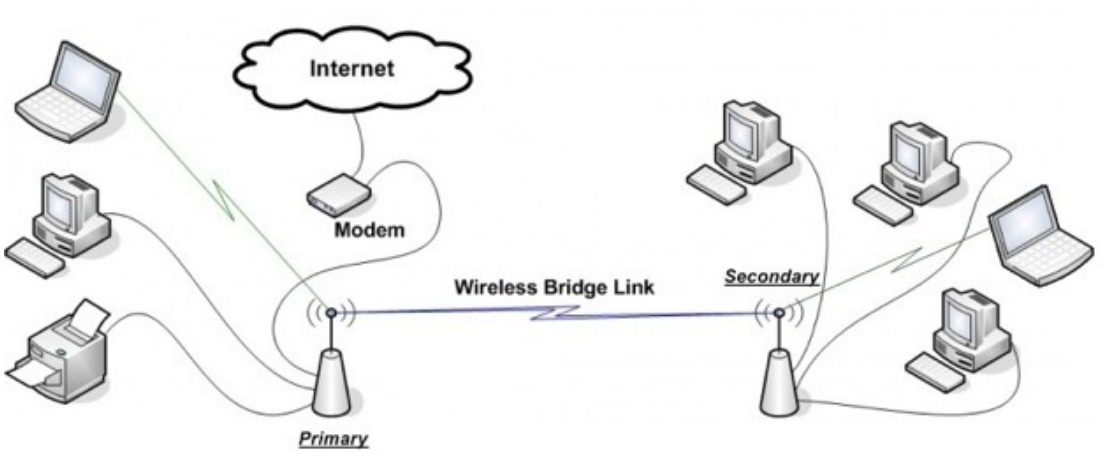

Figure 18. Wireless Repeater (From: 32)

Setting up the wireless bridge repeater can be done using the web based GUI. Configuring the routers does require some extra steps due to known issues in the wireless bridge repeater firmware code [32]. To overcome these issues, the router must first be configured as a wireless bridge prior to configuring it as a wireless bridge repeater. Attempts to configure the router as a wireless bridge repeater without first configuring it as a wireless bridge resulted in failure of the wireless interface. This results in a packet drop rate of 100%.

Once the router is configured as a wireless bridge it could then be configured as a wireless bridge repeater. Figure 19 shows the initial wireless bridge repeater set-up. Both end user devices are able to communicate with each other and remotely access the router configuration GUI. Detailed instructions on configuring the routers as wireless bridge repeaters can be found in [31] [32].

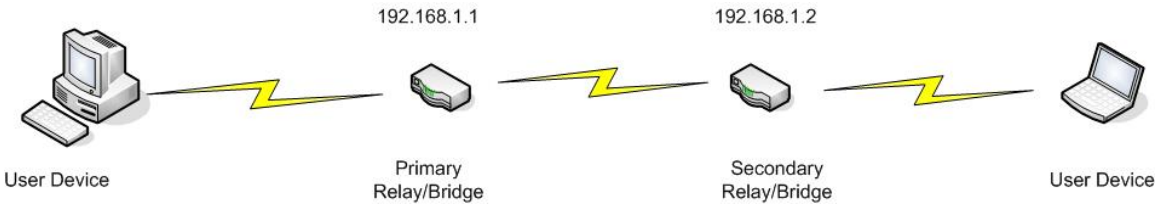

Figure 19. Wireless Bridge Repeater Set-Up

c. Packet Sniffer

In order to implement an Adaptive Bandwidth algorithm on the WRT54GL routers, we first have to find a way to measure traffic characteristics at the outgoing interface. By measuring the inter-transmission times of packets on the port servicing the sensitive network traffic, we can determine packet jitter. A basic packet sniffing program was written in the C programming language and deployed on the WRT54GL routers. Using the open source library *libpcap* we were able to capture outgoing traffic. The timestamp associated with the packet is stored in memory until a user defined number of packets have been captured. Once the defined number of packets has been captured, their timestamps are used to compute the average jitter over the windowed period of time. This average is compared with prior averages and if it exceeds the threshold for that type of traffic, the Adaptive Bandwidth Algorithm is notified.

d. Adaptive Bandwidth Algorithm

Dynamically adjusting the bandwidth allocated to jitter and delay sensitive applications, such as VOIP and Streaming Video, will provide a higher QoS level to end users. Using Linux Shell Scripts and the Packet Sniffer described above, the Adaptive Bandwidth Algorithm is written and deployed on both WRT54GL routers. Basic routing functionality is controlled by the underlying firmware, in this case the DD-WRT firmware package. The Shell script is automatically executed when the router is started. When packets arrive at the input buffer the script marks them for future QoS handling based on their input port. Table 3 provides the incoming and outgoing port assignments for the possible data streams used in the network. After being prioritized based on the desired queuing scheme, the packets are transmitted to the next node on the assigned transmission port.

When the packet on a port of interest is transmitted on the *eth1* interface, it is captured by the packet sniffer deployed on the router. Using the time of the capture, the inter-transmission time between packets is determined and stored in an array. The adaptive algorithm, shown if Figure 20, uses a windowing method to determine the average jitter in the network. The length of the window is defined in the script and can be

adjusted by the user. The current setting is a window length of 200 packets. When the array fills, the average delay is calculated for that period. Comparing this value with prior values allows the script to determine network jitter. If this jitter exceeds a threshold value, currently set at 100ms, the variable controlling the bandwidth apportionment for the application is increased. This procedure continues until all of the bandwidth has been assigned to the application.

During the course of developing the network, multiple implementations of the algorithm were examined. Implementations ranged from Linux Shell scripts to self contained C programs. In each case compatibility problems arose between the software and the routers operating system. These issues caused the router to fail less than fifteen seconds into each test run requiring the firmware to be reinstalled. These issues have not been resolved at the conclusion of this thesis and remain for future development.

	Receiving Port Range	**Transmission Port Range**
VOIP	10000-10499	10500-10999
Audio	11000-11499	11500-11999
Video	12000-12499	12500-12999
Light Sensors	13000-13499	13500-13999
Temperature Sensors	14000-14499	14500-14999

Table 3. Linksys WRT54GL Input/Output Port Assignments by Application

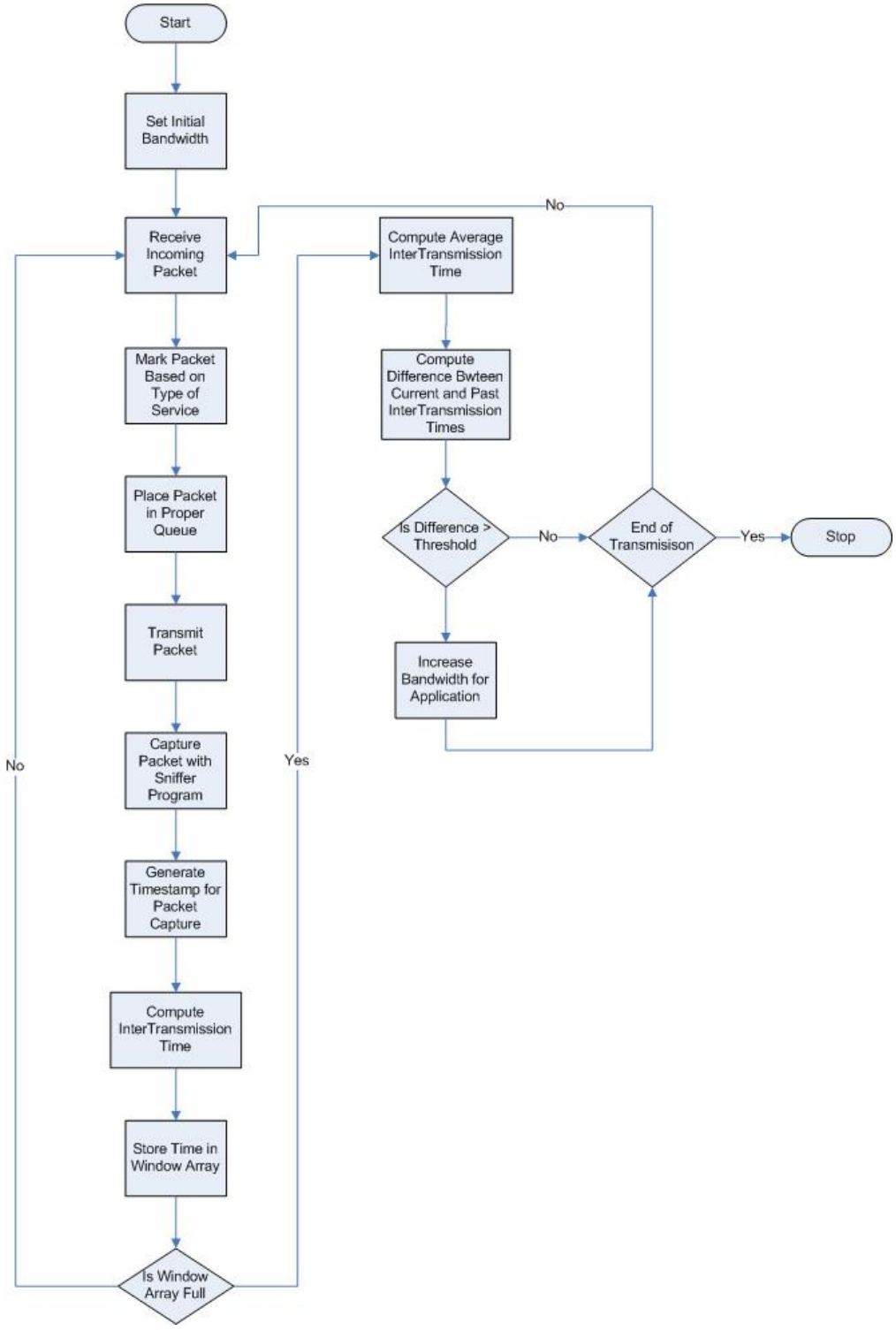

Figure 20. Adaptive Bandwidth Algorithm Block Diagram

3. Sink Node

a. Packet Translator

Data packets generated by the periodic sensor nodes, Sun SPOTs, are transmitted to the sink node using the IEEE 802.15.4 protocol. The transmission range of the SPOT is limited to 100m outdoors and considerably less indoors [33]. In order for data provided by the SPOTs to reach the end users, a more powerful and robust transmission protocol must be used. All of the components from the sink node to the end user devices utilize the IEEE 802.11 protocol.

Before the JAVA ME datagrams generated by the SPOTs can be translated across the IEEE 802.11 segment of the network, they must be converted into JAVA SE datagrams. A software translation program was written and deployed on the sink node. The sink node serves as a gateway between the two segments and is the ideal location for the translator. The sink maintains both an IEEE 802.15.4 connection with the sensor field through the SPOT basestation and an IEEE 802.11g connection with the relay node, e.g. WRT54GL router, via the installed Airport Extreme wireless network card.

The translator program receives the incoming IEEE 802.15.4 and stores the contents in memory. The contents are then converted into Strings in order to comply with the JAVA SE datagram structure. These Strings are loaded into a byte array, which is then encapsulated in the JAVA SE datagram. The program then transmits the datagram across the IEEE 802.11 segment using a Datagram Socket connection with the target device. During this initial development phase of the network, the target address is a static variable configured by the user. Figure 21 shows a block diagram of the translator program.

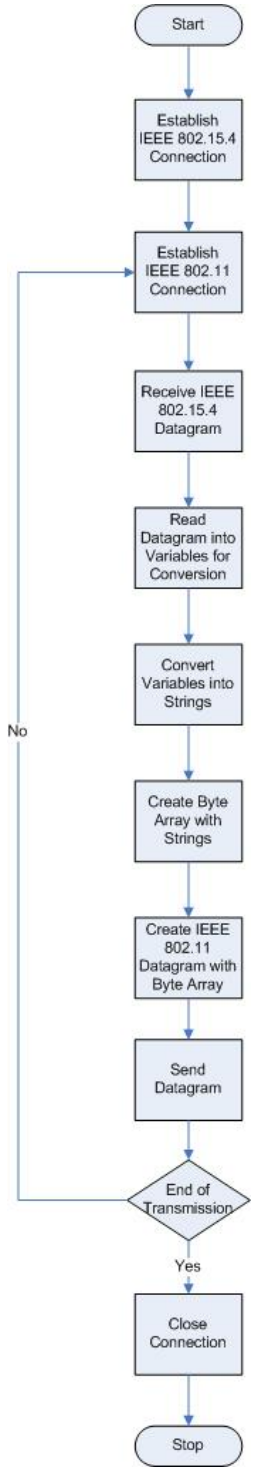

Figure 21. IEEE 802.15.4 Datagram to IEEE 802.11 Datagram Translator Block Diagram

b. Adaptive Sensor Sampling Algorithm

Quality of service in the network begins with the sensors themselves. If the quality of the network is diminished during the initial stages, the degradation will compound until the network is unusable. Two of the factors that can effect the performance of the network are node density and transmission rate. Node density in the network is a function of the mission and network design and developers have little control over the number of sensors that will be deployed in any given network. With increased node density comes greater contention in the transmission channel, which results in decreased network performance. Transmission/sampling rate can also play a critical role in network performance. Increased transmission rates result in more packet collisions which result in decreased performance. By monitoring the delay experienced at the sink node, the sampling rate of the sensors can be decreased to improve performance.

The Sun SPOTs used in the network are programmable and reconfigurable. A program was written that monitors the inter-arrival jitter between the packets at the sink node. If the jitter exceeds a user defined threshold the sink node sends a notification to the sensor. Upon receipt of the packet the sensors adjust their sampling rate to reduce the amount of jitter. This process continues until the performance of the network is stable. Figure 22 shows a block diagram of the algorithm developed.

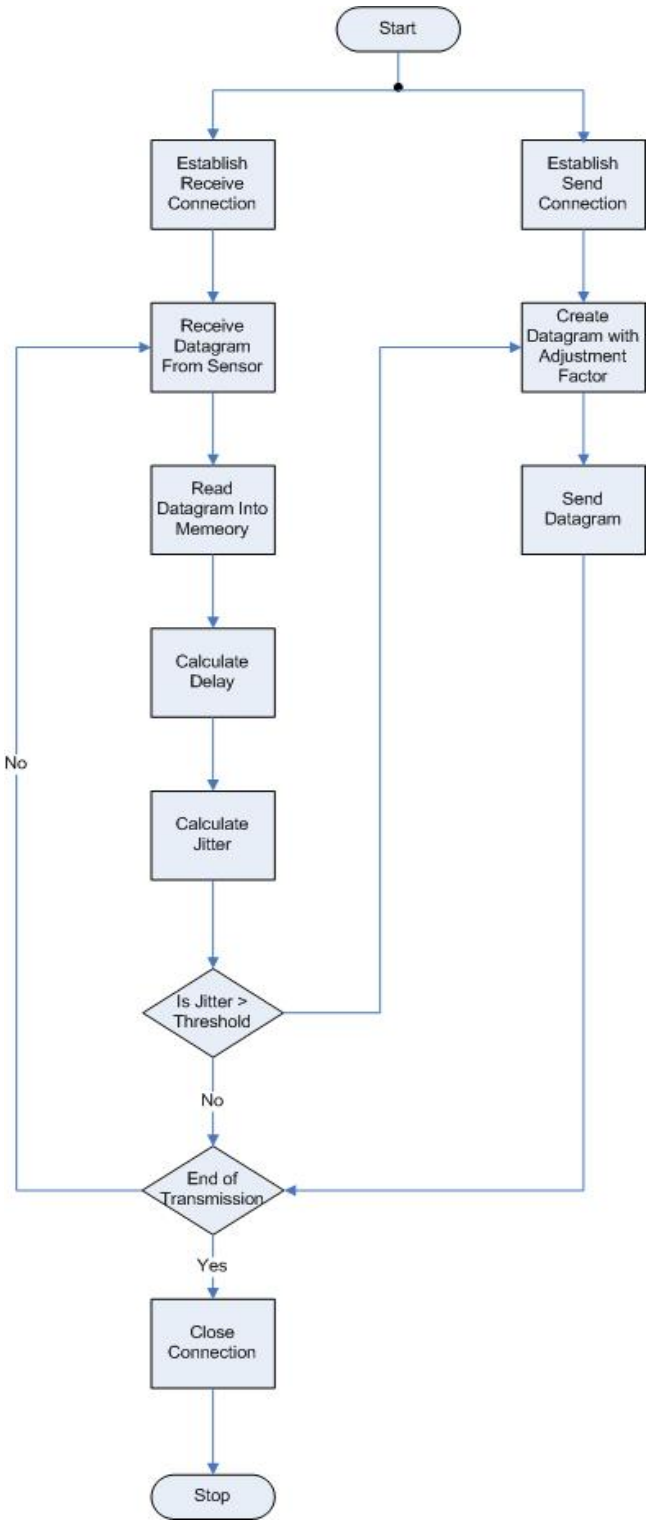

Figure 22. Adaptive Sensor Sampling Algorithm Block Diagram

C. SUMMARY

This chapter introduced the network prototype. Components used in the development were presented and examined. Three components were chosen to serve as the end user devices for the network. The desktop PC was used to simulate equipment available at a headquarters element while a laptop was used to simulate equipment available in a vehicle. The final device examined was a small Internet tablet which could be easily carried by personnel in the field. A user interface was created for the PC and laptop to enable user to monitor the network.

A detailed examination of Linksys WRT54GL router was also provided. The router will serve as the long haul backbone of the network, providing wireless communications over an IEEE 802.11g link. Because the router is a critical segment of the network, a detailed analysis of the functionality of the router was conducted. The Linksys firmware was determined to be inadequate for the concept of the network and a third-party package from DD-WRT was utilized. An attempt to implement an Adaptive Bandwidth Algorithm on the router was examined and the program was determined to be incompatible with the firmware.

Finally, the sink node and sensor fields were examined. Sensors used in the development of the network included Sun SPOT sensor motes and DCS-5250 Internet Video Cameras. The Nokia N80 Internet Tablet originally intended to serve as the sink node was determined to be ineffective. The lack of JAVA compatibility and the limited capabilities of its operating system hindered development. In order to continue developing the network, a laptop computer was used as a substitute.

Chapter IV presents the experimental design that will be used to evaluate the performance of the network. Performance metrics are introduced and equations are developed to evaluate the network. Models used to generate multiple types of traffic in the network are developed and presented. The chapter concludes with an overview of the experimental set-up is used to evaluate the performance of the network.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. EXPERIMENTAL DESIGN

This chapter covers the methodology and design of the experiments that are used to evaluate the performance of the network. The chapter begins by identifying the performance metrics that will be used to evaluate the network's performance. This is followed by a description of the models used to simulate the various types of traffic in the network. The chapter concludes with a description of the set-up and procedures used in each experiment.

A. PERFORMANCE METRICS

1. Packet Loss

Packet loss is a measure of the percentage of packets lost in transmission. This can result for a number of reasons such as network congestion, network topology changes, packet size and channel interference. Packet loss can be measured by comparing the total number of packets transmitted from the nodes with the number of packets received at the End User Device (EUD). The average percentage of package loss is given by:

$$\text{Avg. \% Packet Loss} = \frac{\sum_{i=1}^{\text{No. of Trials}} \left(\frac{\left(\sum_{j=1}^{\text{No. of Nodes}} \text{No. of Pckts Trans. by Node j} \right) - \text{No. of Pckts Rcvd at EUD}}{\sum_{j=1}^{\text{No. of Nodes}} \text{No. of Pckts Trans. by Node j}} \text{ X } 100 \right)}{\text{No. of Trials}} \quad (0.1)$$

2. Latency

Latency is the time delay between the start of packet transmission and the start of reception. Latency can be measured as either a one-way, end-to-end, value or as a round-trip value. In a typical network, the overall latency experienced is a combination of many factors. As the distance between nodes increases the delay incurred due to propagation through the channel also increases. Additionally, at each node along the transmission path

the packet can experience delay from factors such as queuing and processing. End-to-end latency can be measured by comparing the transmission time of a packet with the reception time. The average latency is given by:

$$\text{Avg. Latency} = \frac{\sum_{i=1}^{\text{No. of Trials}} \left(\frac{\sum_{j=1}^{\text{No. of Packets}} \text{Time of Reception - Time of Transmission}}{\text{No. of Packets}} \right)}{\text{No. of Trials}} \quad (0.2)$$

3. Jitter

Jitter in the network is defined as the variation in end-to-end latency of the received packets. Figure 23 shows a graphical depiction of jitter in a network. The sending node on the left-hand side transmits packets in a continuous stream with relatively even spacing between the packets. As the packets travel through the network and begin to experience congestion, the spacing between the packets begins to vary and packets clump together. At the receiving node, the latency between the received packets is no longer constant. Packets which clumped together can have the same end-to-end latency while other packets can have a significant variation in their end-to-end latency. Some applications, such as VOIP, are highly susceptible to jitter. Variations in packet arrival can cause the conversation to appear choppy. In a network, this can be measured by calculating the latency for each received packet and comparing the variation. The average jitter is given by:

$$\text{Avg. Jitter} = \frac{\sum_{i=1}^{\text{No. of Trials}} \left(\frac{\sum_{j=1}^{\text{No. of Packets-1}} |\text{Latency pckt}_j - \text{Latency pckt}_{j+1}|}{\text{No. of Packets-1}} \right)}{\text{No. of Trials}} \quad (0.3)$$

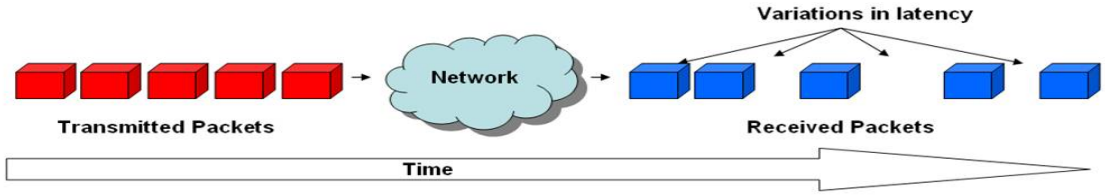

Figure 23. Network Jitter.

B. MODELING OF DATA STREAMS

Sensor networks can be designed for a wide range of user defined applications. Their low-cost allows them to be both reconfigurable and scalable. Since it would be impossible to test every possible combination of sensors that may be incorporated into a WSN, three types of sensor traffic that would be common in a military WSN were chosen to simulate network traffic; VOIP, Streaming Video and Periodic Sampling (light and temperature sensors) . Streaming video is be provided by the DCS-5220 Internet video cameras. The remainder of the network traffic is simulated.

1. Voice Over IP (VOIP)

VOIP is one of the most restrictive applications that can be deployed in the network. The application is bandwidth intensive and requires compliance with strict tolerances and stability. High packet drop rate, excessive latency and excessive jitter can result in a conversation that is unintelligible. Because the need to communicate on the battlefield is of importance, the network is being designed with the future integration of VOIP in mind. Since no actual VOIP applications are available during this phase of the design, VOIP traffic has to be modeled.

To model VOIP traffic a generation scheme based on the ITU-T four state continuous time conversational model [34] is used; the model is illustrated in Figure 24. The model shows the four states a conversation can be in, two states are for a single participant talking at a time, one state for no participants talking and the final state for

both participants talking. At the end of each cycle the probability of changing states is determined and the state of the conversation adjusted accordingly. This cycle repeats until the conversation is terminated.

Because the traffic is IP based, the ITU-T continuous time model has to be adapted to generate packets in discrete time [35]. Using the four state model as guidance, an algorithm is designed to simulate the conversation between two individuals based on probability of packet transmission. Based on the probability of transmission, each node determines if it will transmit a packet during the cycle based on its starting state.

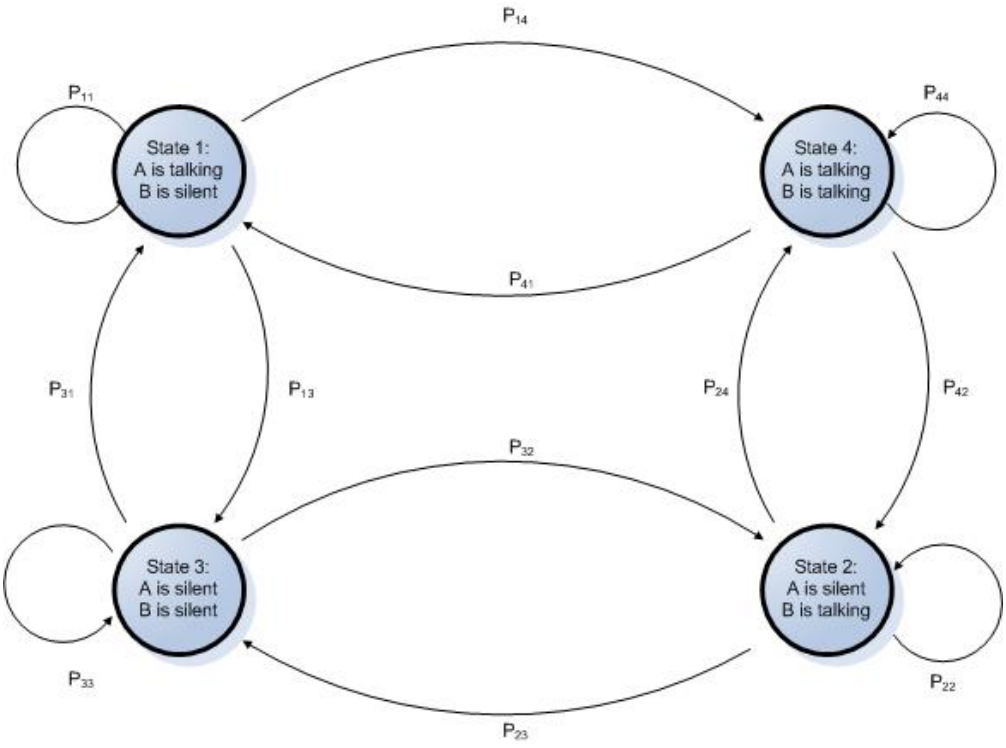

Figure 24. Voice-Over-IP Discrete-Time Conversational Model. (From: 34)

Figure 25 shows the block diagram of the algorithm used to generate traffic as implemented on the node which initiated the call. The call initiator begins the conversation in state 1. The call is stated by sending a packet to the call recipient. The initiator then determines is it will change states or remain in state 1. Based on the results the initiator can either be in state 3 or in state 1/4. If the initiator is in state 1/4 it must determine which of these states it is in. This is done by waiting to see if a packet was

received from the call recipient. If a packet was received, the initiator changes to state 4, if not, it remains in state 1. This process of determining state based on probability of state change and whether or not a packet was received for the recipient continues until the conversation is ended. A similar model is used by the recipient to determine its state.

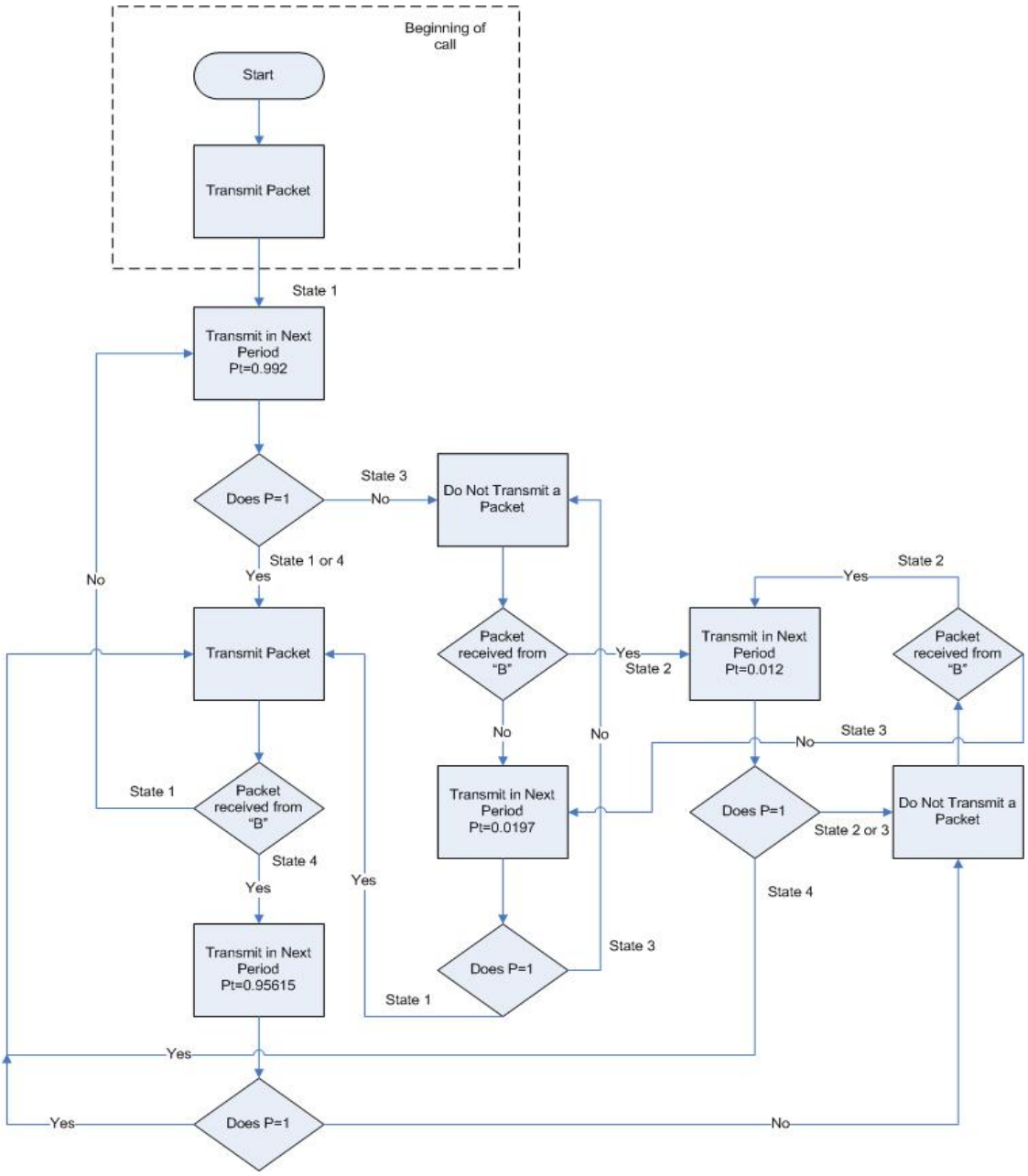

Figure 25. VOIP Conversation Model Block Diagram (Initiator).

In the ITU-T model [34] as shown in Figure 24, the probability of changing states between state i to state j is denoted by P_{ij}. T_{interval} represents the interval between successive cycles and is dependent on the audio codec used. For the purposes of this thesis the G.711 codec was used due to its wide use in VOIP applications. Following the recommendations of G.711 [36], T_{interval} was chosen to be 20ms. $T_{\text{talk-average}}$, $T_{\text{stop-average}}$ and $T_{\text{double-average}}$ represent the average duration a node remains in States 1/2, 3 and 4 respectively. Using the guidance of ITU-T Recommendation P.59, the values used were 1.004 seconds, 0.508 seconds and 0.228 seconds respectively [34]. $K_{\text{talk-double}}$ represents the probability of a node in State 1 or 2 changing to State 4. The parameter is once again based on measurements of human conversation provided by P.59. Using this guidance, the parameter was set at 0.6 [34]. Using equations 4.4 through 4.10, with parameters provided, the probability of a node changing between states was calculated and the resulting probabilities are shown in Table 4.

$$p_{11} = p_{22} = 1 - T_{\text{interval}} / T_{\text{talk-average}} \tag{0.4}$$

$$p_{33} = 1 - T_{\text{interval}} / T_{\text{stop-average}} \tag{0.5}$$

$$p_{44} = 1 - T_{\text{interval}} / T_{\text{double-average}} \tag{0.6}$$

$$p_{31} = p_{32} = \left(1 - p_{33}\right) / 2 \tag{0.7}$$

$$p_{41} = p_{42} = \left(1 - p_{44}\right) / 2 \tag{0.8}$$

$$p_{14} = p_{24} = \left(1 - p_{11}\right) K_{talk-double} \tag{0.9}$$

$$p_{13} = p_{23} = \left(1 - p_{11}\right)\left(1 - K_{talk-double}\right) \tag{0.10}$$

		Ending State			
		1	**2**	**3**	**4**
Starting State	**1**	0.97922		0.03463	0.01247
	2		0.97992	0.03463	0.01247
	3	0.01969	0.01969	0.96063	
	4	0.04356	0.04356		0.91228

Table 4. VOIP Conversation Model Probability of State Change

2. Periodic Sampling

Periodic sampling devices are the one of the most common devices in use in WSNs. These devices sample the desired phenomenon at regularly defined intervals. While some periodic sampling devices, such as control sensors, have strict latency requirement, for the purpose of this thesis the periodic data streams are assumed to have no latency or jitter requirements.

Two types of periodic sampling data streams were utilized in the development and testing of the network, a temperature sensor data stream and a light sensor data stream. Each stream produced periodic packets of a fixed size, 77 bytes for the light sensor datagram and 81 bytes for the temperature sensor datagram. Figure 26 shows the content of the light datagram. Each packet contains a standard IEEE 802.15.4 49 byte header [37]. The header is followed by an 8 byte segment which contains the unique 64-bit identification number for each sensor node. The sequence number segment contains an 8 byte integer, which identifies where in the sequence the datagrams belong. This is followed by a timestamp that identifies the system time when the datagram was created. System times on the sensor nodes are synchronized with the sink node when each node is loaded with its desired application. The final segment of the datagram is the actual sensor

reading. In the case of the light sensor, it is a 4 byte long integer value. The segment for the temperature sensor is twice as long as the light sensor, 8 bytes versus 4 bytes, because the value produced is a double.

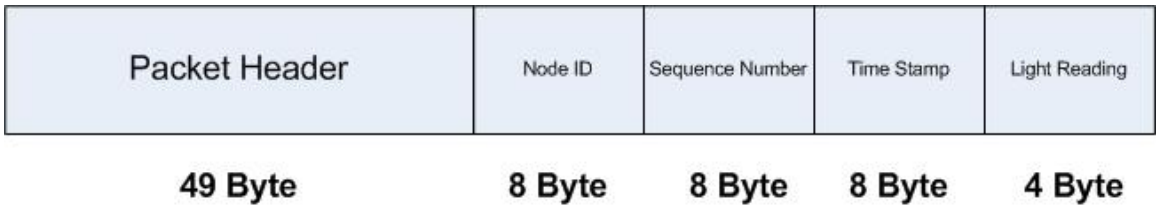

Figure 26. Periodic Light Sampling Datagram Packet

Both the temperature and light sensor data streams follow the same model. The only difference between the two models is the content and length of the final segment of the datagram payload. Figure 27 shows a block diagram of the periodic sampling application used in the development and testing of the network. The first step in the model is to establish a connection between the SPOT and the sink node. The SPOT opens a broadcast connection on the desired port while the sink node opens a server connection on the same port. This allows the SPOT to transmit to anyone listening on that port while the server connection allows the sink node to receive any packets transmitted on the port. After setting the time stamp and sampling the desired phenomena, a datagram is created which contains the sending nodes identification number, the sequence number of the packet, the time stamp and the collected sensor reading. The datagram is transmitted on the previously established broadcast connection. After transmission the model enters a waiting loop until the sampling interval has expired at which time the cycle repeats.

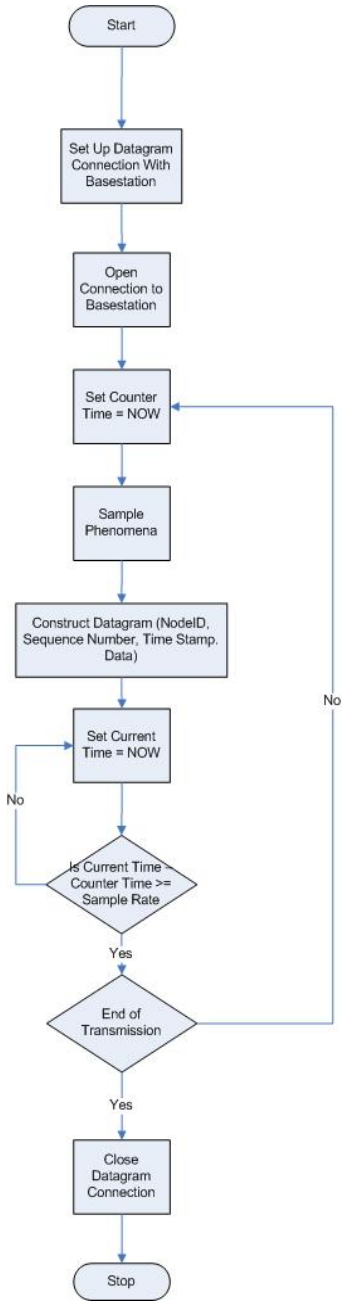

Figure 27. Period Sampling Model Block Diagram

C. EXPERIMENTAL DESIGN AND SET-UP

1. Sensor Field Characterization

The sensors provide the primary functionality of the network. In this network periodic sampling data streams are provided by Sun SPOT sensor motes. To understand the performance characteristics and limitations of the SPOTs a series of experiments were conducted to measure packet latency, jitter and the packet drop rate.

The series of experiments involved one SPOT basestation, six SPOT free range motes and a MacBook Pro laptop. The free range SPOTs were arranged in a mesh network topology with 30 inches separating each mote and the basestation. The basestation was connected to the laptop using a USB 2.0 cable. Output from the system was printed to the user interface on the laptop. The number of free range SPOTs turned on was increased from one to six during each successive series of experiments. Each SPOT provided a periodic light sampling data stream. Measurements were taken with sampling rates of 100ms, 200ms, 500ms and 1000ms for five minute periods. The experiment for each configuration, number of active SPOTs and sampling interval, was repeated five times to obtain an average of the results. Figure 28 shows the experimental set-up used to characterize the sensor field.

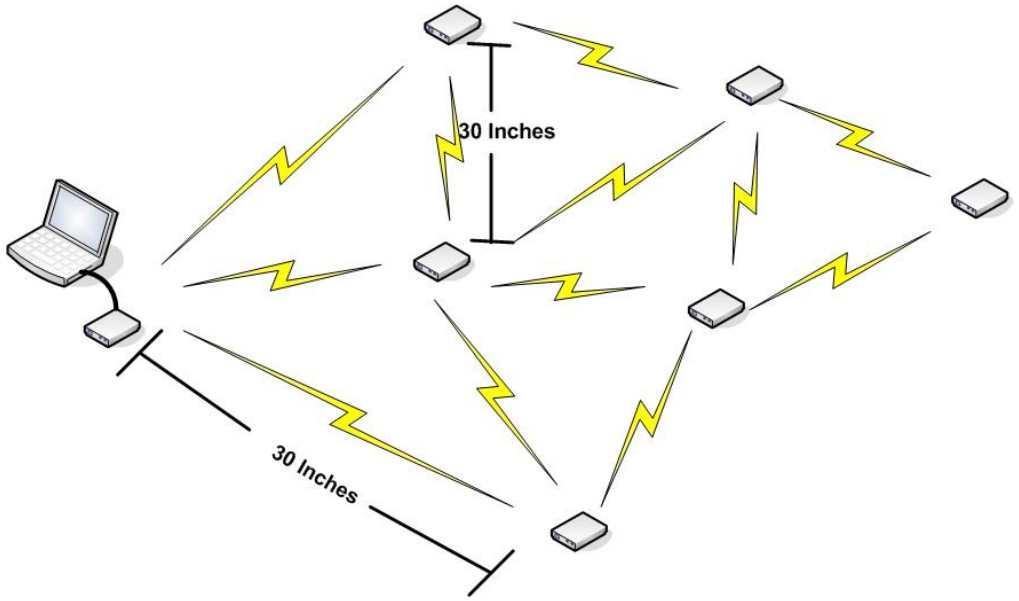

Figure 28. Sensor Field Characterization Experimental Set-Up

2. Adaptive Sensor Sampling Performance

As the scale of the network increase, contention amongst sensor nodes increases. The adaptive sensor sampling algorithm, adjust the sampling rate of the sensor nodes in real time. The algorithm measures the delay experienced by each packet at the sink node. Packet delay is monitored for a predefined number of packets after which the average delay for the windowed period is calculated. This average delay is compared to a maximum and minimum threshold delay value. If the average delay exceeds the maximum threshold value a packet is sent to the transmitting node instructing it to slow its sampling rate by a predefined adjustment factor. If the delay average is below the minimum threshold value the transmitting node is instructed to increase its sampling rate by the adjustment factor. If the average delay is between the threshold values a packet is sent to the transmitting node instructing it to maintain its sampling rate. To measure the effectiveness of the algorithm, the same basic set-up, Figure 28, and procedure used in the sensor field characterization experiments were used. The initial sampling rate was set to 100ms. The threshold level for the adaptive algorithm was set to 200ms with an

adjustment rate of ±50% of the current sampling rate. Results of the experiments were compared with those from the initial sensor field characterization experiments to determine performance results.

3. Single Router Performance

Because the concept of this network is to deploy sensors in remote locations, a stronger signal than that provided by the SPOTs must be used to reach the end user. In this network, the data generated by the sensor field is relayed from the sink node to the end user over a stronger IEEE 802.11g wireless link using a Linksys WRT54GL router. The links between the router and the sensor field and end users is the longest link in the network. As such, it has the potential to provide the most adverse impact on the overall performance of the network. To evaluate the impact of this link, the end-to-end performance of the network with the router was measured and compared with the results from the sensor field characterization.

In the basic test configuration, only one router was placed between the sink node and the end user device. This router was considered the primary router for the remainder of the experiments. In the case of this series of experiments, the end user device was an HP desktop computer while the sink node was a MacBook Pro laptop. Five sun SPOT free range sensor motes, separated by 30 inches, were used to generate periodic light samples at intervals of 500ms. Data packets generated by the SPOT were passed to the sink node where they were translated from the IEEE 802.15.4 protocol to the IEEE 802.11g protocol. The sink node then passed the translated packets through the relay/bridge node to the end user device. The relay bridge node was separated from the sink node and end user device by 20 feet in each direction. To measure the performance, the time stamp in the datagram was compared with the system time stamp at the receiving terminal. Differences were used to compute the end-to-end latency and jitter. Using the sequence number contained in each packet, the total number of packets generated was compared with the number of packets received to determine the packet drop rate. Figure 29 shows the basic test set-up used to evaluate the impact of a single relay/bridge node.

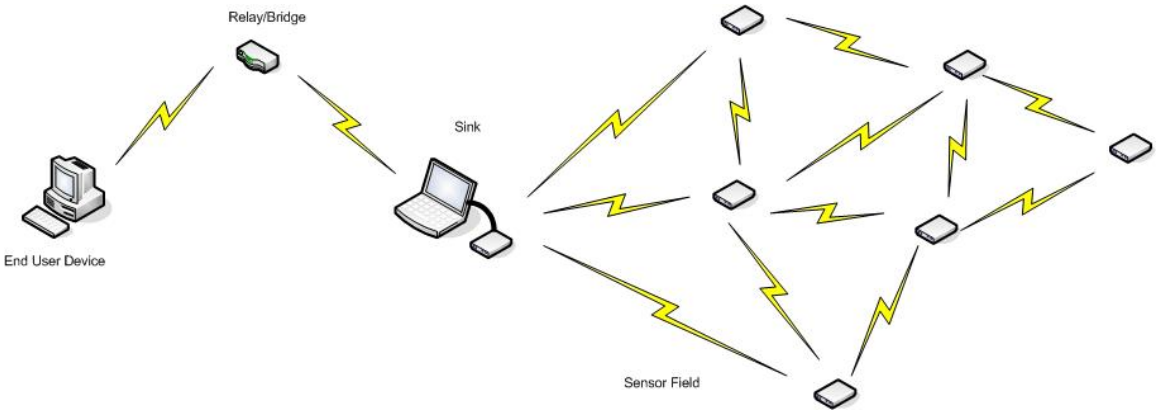

Figure 29. Single Router Performance Experimental Set-Up

4. Bridged Two Router Performance (Single Sensor Field)

To extend the range of the network, the concept calls for the use of a remote control helicopter with a router mounted to the bottom to serve as an airborne relay/bridge node. To evaluate the impact of this additional node on the performance of the network, the series of experiments conducted for the single router were repeated with the additional router placed between the primary router and the end user device to simulate the airborne node. This secondary router was considered the secondary router for the remainder of the experiments. The distance between the sink, routers and end user device was maintained at 20 feet between each node. All other experimental parameters were held constant. Figure 30 shows the basic test set-up used in the experiments.

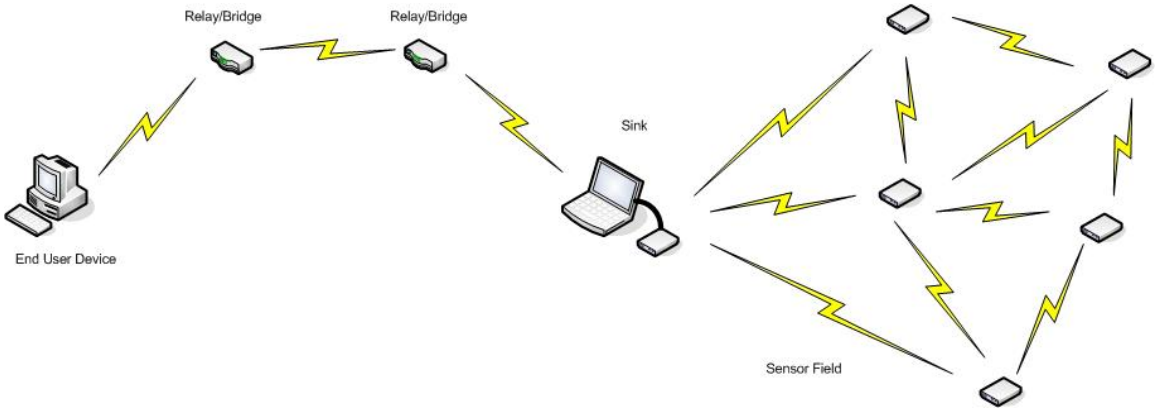

Figure 30. Bridged Two Router Performance Experimental Set-Up

5. Bridged Two Router Performance (with VOIP and Streaming Video)

Since this network is being designed as a multi-media WSN, the performance of the network was evaluated using multiple data streams of varying packet lengths and transmission rates. The basic test set-up remains the same in the two router performance experiments. Two additional data streams were added to the network to increase the loading on the IEEE 802.11 links. Using the model outlined in section III.B.1, a simulated VOIP conversation was generated between a MacBook Pro laptop, connected to the primary router, and the desktop computer, i.e. end user device. A streaming video stream was also added to the network load at the primary router using the DCS-5220 Internet video cameras. All other experimental parameters were held constant. Figure 31 shows the experimental set-up used to evaluate the effect of increased loading on the network's performance.

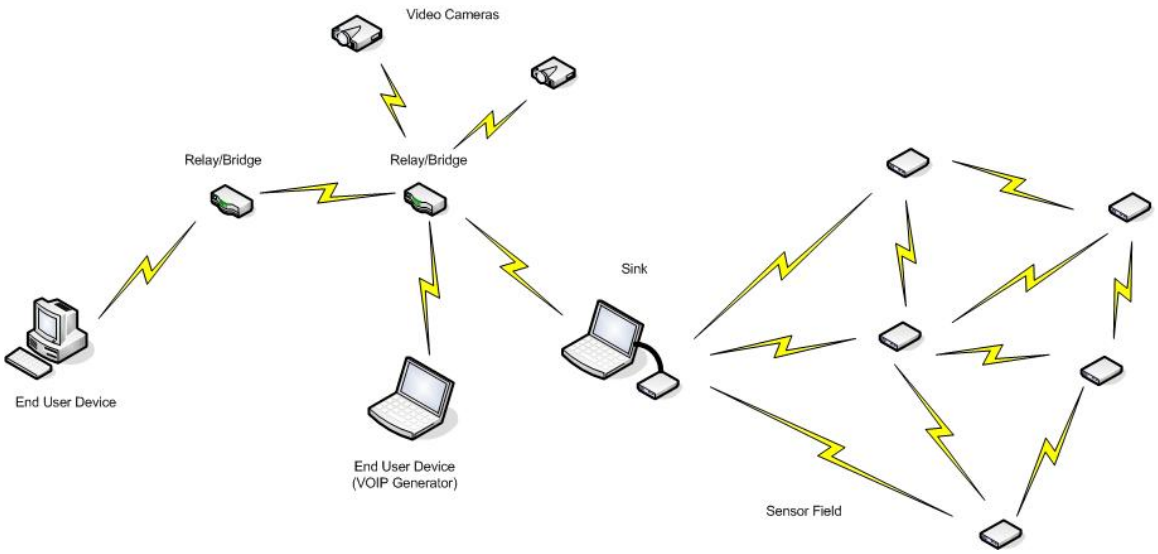

Figure 31. Bridged Two Router Performance Experimental Set-Up

6. Adaptive Bandwidth Performance

Because the IEEE 802.11 links experience the heaviest loading, the performance of the primary and secondary router is key to the end-to-end performance of the network. To examine the impact of the Adaptive Bandwidth Algorithm, the basic set-up used to

evaluate the bridged two router network (with VOIP and streaming video), Figure 31, was utilized. The Adaptive Bandwidth Algorithm was implemented on both the primary and secondary routers. The inter-transmission threshold level of the algorithm was set to 90 ms with an adjustment factor of ±10% of the current bandwidth.

D. SUMMARY

This chapter covered the methodology that was used to evaluate the performance of the network. The network performance metrics of packet latency, jitter and drop rate were defined and equations for their calculations presented. To test the performance of the network, data streams of varying packet length and transmission rate needed to be passed through the network. Two models for generating data streams were presented, VOIP and Periodic Sampling. Finally, the experimental set-up and parameters used to test the performance of the network were discussed. These experiments examine the performance of the network at each stage of the development and integration of the network.

Chapter V discusses and analyzes the experimental results. It also provides recommendations for network configuration.

THIS PAGE INTENTIONALLY LEFT BLANK

V. EXPERIMENTAL RESULTS AND ANALYSIS

This chapter presents the results of the network performance experiments. Key network performance characteristics of packet latency, jitter and drop rate are examined at each phase of development. Analysis and recommendations are made as to optimize network configurations.

A. PACKET DROP RATE (PDR)

Packet drop rate is one of the basic measures of performance in a network. While some networks and applications, such as non-real time environmental monitoring, are not sensitive to excessive packet delay or jitter, all applications are sensitive to the loss of too many packets. PDR will be examined at each stage in the development of the network. The first series of experiments will examine the performance characteristics of the Sun SPOT sensor field and basestation. This will be followed by measurements examining the impact of adding wireless relay/bridge nodes into the network. The PDR discussion will conclude with a look at the performance results of an Adaptive Sampling Algorithm that was applied to the sensor field.

As data packets are collected and transmitted from the Sun SPOT sensors, they must pass through the Sun SPOT basestation to reach the sink node. Because of the limited capabilities of the basestation, this is the first potential area for significant degradation of the data stream. The first series of trials evaluates the PDR of the sensor field at various sampling rates and with varying numbers of nodes connected. To examine the effects of increased node density in the network, the sampling rate is held constant and an additional SPOT is added to the network for each series of trials. The number of SPOTs used varies between one and eleven SPOTs in a two square meter area. This procedure was then repeated for other sampling rates.

Examining the results shown in Figure 32, it is clear that at each sampling rate there is a critical node density, beyond which PDR increases significantly. With only one SPOT connected to the network, the PDR is relatively constant at approximately 1.46%.

Increasing or decreasing the sampling rate has no significant impact of the PDR. As the number of nodes connected to the network is increased, sampling rate becomes a key variable in the performance of the network. With two Sun SPOTs connected to the network, only the fastest sampling rate, 1 sample/100 ms, is significantly impacted. This is the first critical node density identified. With the addition of the second node, the PDR at 100 ms sampling rate increased from 1.5% to just over 20%. Adding a third node at 100 ms caused the PDR to increase more than 20% to approximately 45%. While the percentage of PDR does not increase as much with each successive node added, the trend remains positive until reaching a maximum observed value of approximately 87% at a node density of eleven.

As shown in Figure 32, the second critical point is at three nodes when the sampling rate is 1 sample/200 ms. When sampling at every 200 ms, increasing the node density beyond three results in an increase of 20% at a node density of 4. As with the 100 ms sampling interval, this trend continues upward reaching a maximum of approximately 70% with a node density of eleven. The third and final critical point observed is at a node density of eight when sampling every 500 ms. The increase is more subtle than in the previous two sampling rates. The first increase is less than 10% but the trend does continue to increase as nodes are added.

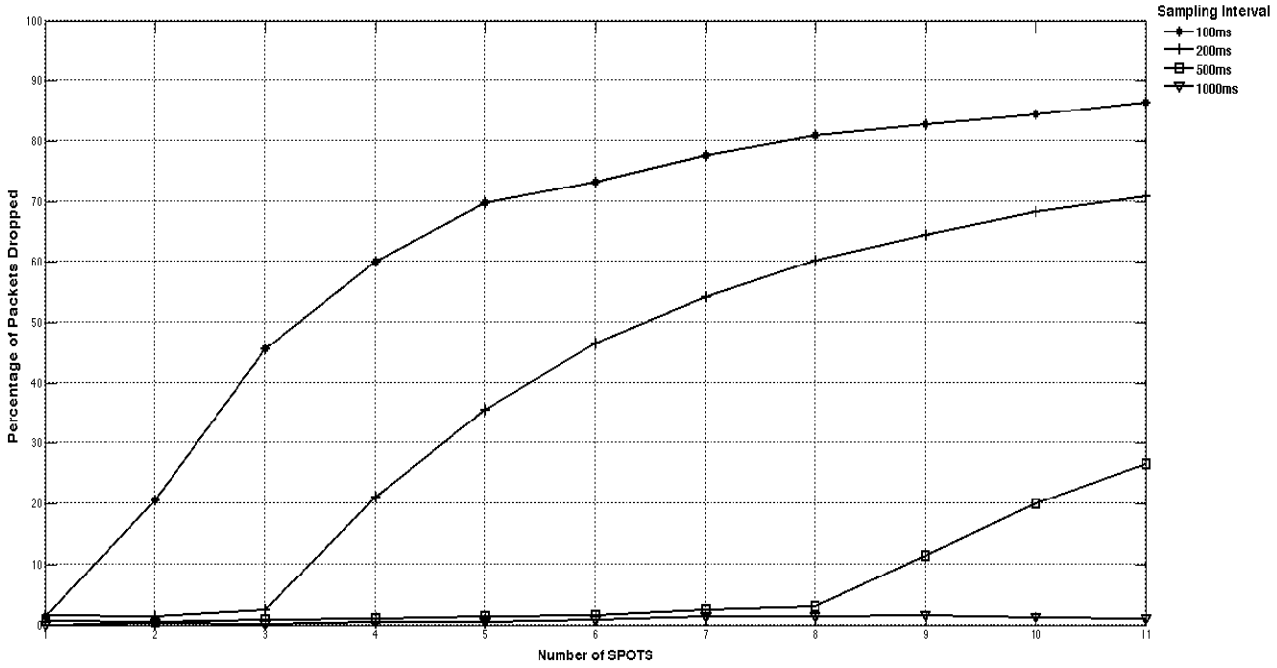

Figure 32. Sensor Field Packet Drop Rate (Sample Intervals 100 ms, 200 ms, 500 ms, 1000 ms)

The only sampling rate which did not exhibit a critical point within the scope of the trials was 1000ms. When sampling at this rate, the PDR remains relatively constant as more nodes are added as shown in Figure 33. This does not imply the non-existence of a critical point at a sampling interval of 1000 ms. Within the limited scope of this network no critical point was observed.

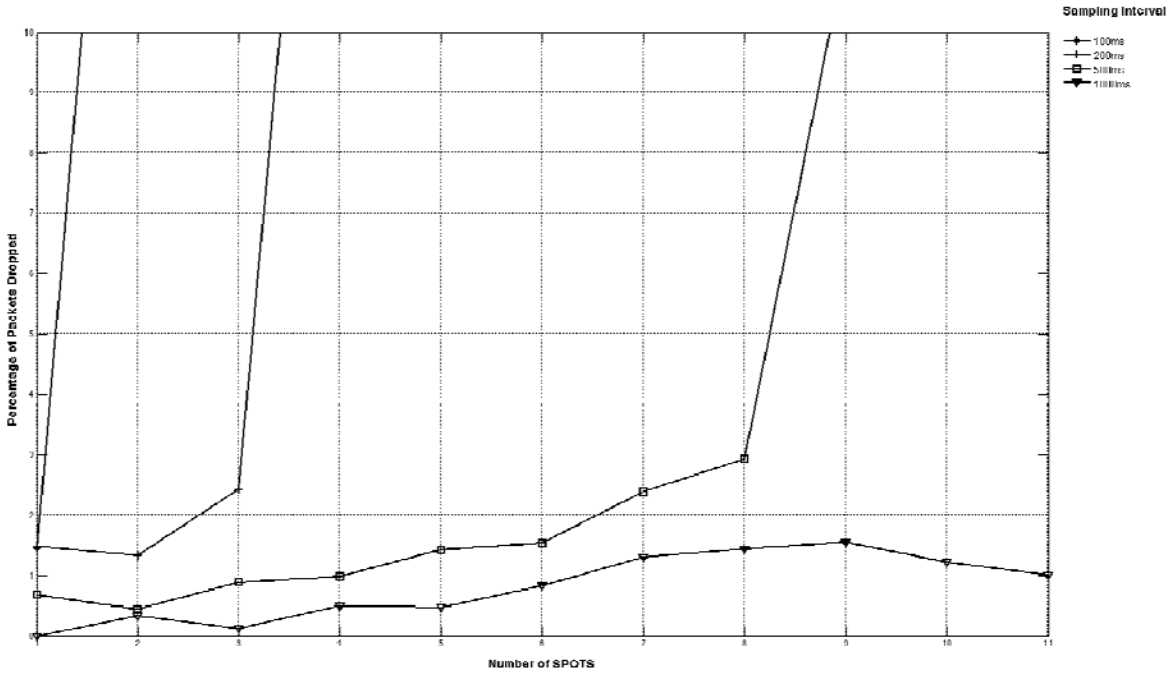

Figure 33. Sensor Field Packet Drop Rate (Enlarged View)

The next segment to examine is the wireless relay/bridge link provided by the WRT54GL router. First, a single wireless router is used to relay the homogenous sensor data stream from the sink node to the end user device. This is then expanded to look at the effects of a two node wireless bridge relay configuration. Figures 34 and 35 show the resulting PDR plots for the two configurations. While the PDR may have increased slightly in each of the configurations, the general shape of the curves and trends remain the same as in the sensor field to sink node curves. The critical points remain at the same locations for the faster sampling rates while the 1 packet/1000 ms sampling rate maintained a relatively constant PDR of approximately 2%.

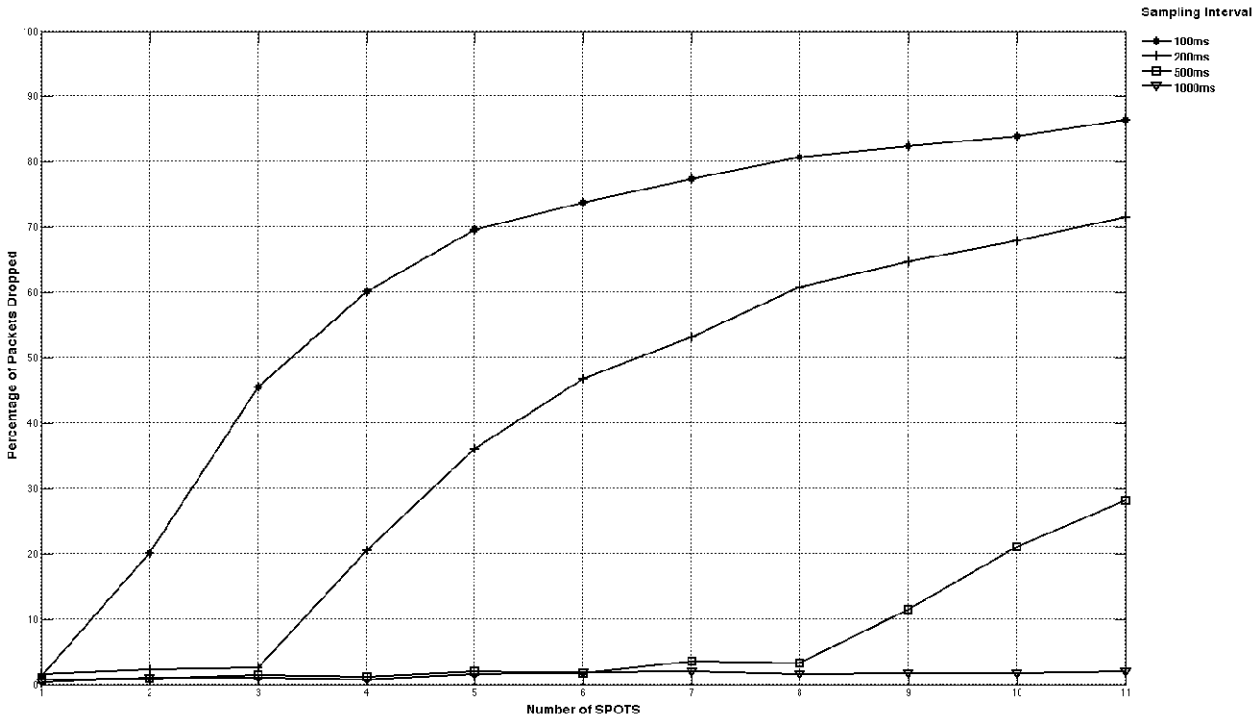

Figure 34. Packet Drop Rate Across A Single Relay Node

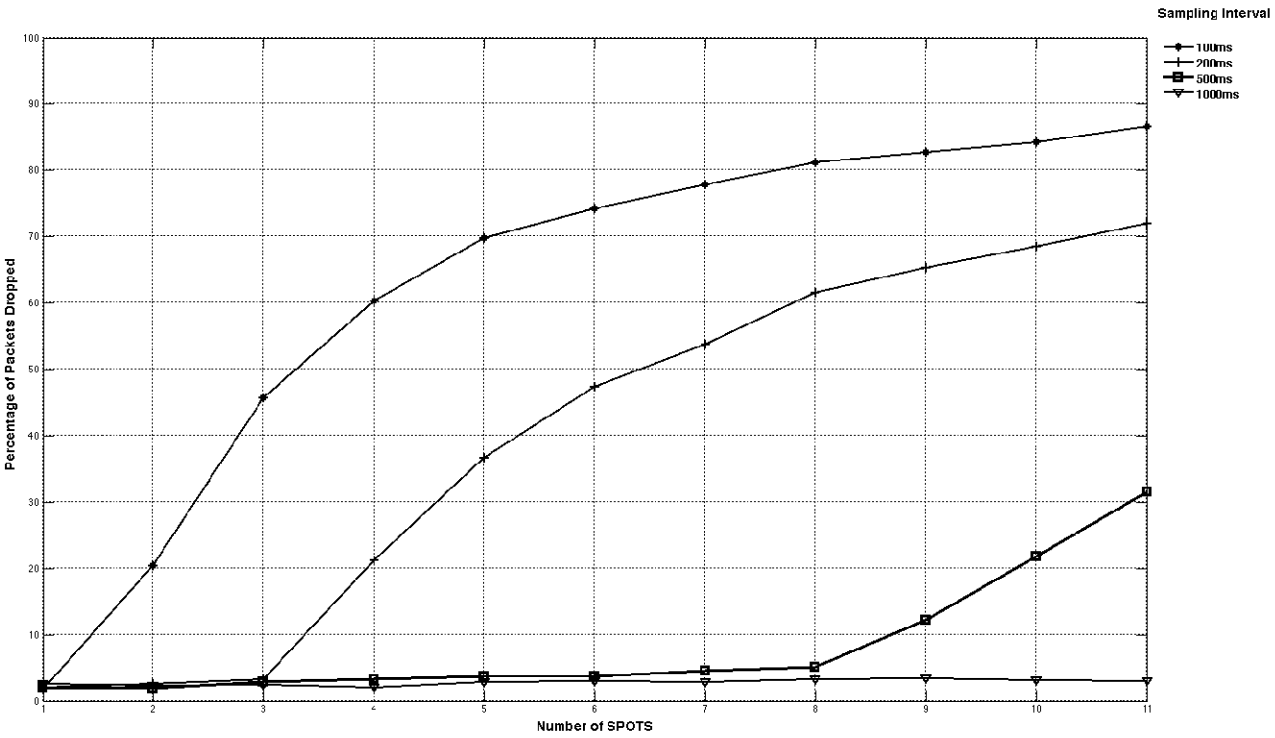

Figure 35. Packet Drop Rate Across A Two Node Relay (Homogeneous Traffic)

The next trial introduces two additional data streams into the network. A streaming video data stream is provided by the DCS-5220 Internet Video Camera and injected into the network at the first router. Additionally, the modeled VOIP signal is also injected into the network at the first router. Figure 36 shows the resulting PDR plot. The curves remain almost identical to those observed earlier. While the additional data streams do increase the network load, it is not enough to stress the network. The capacity of the WRT54GL is capable of delivering the sensor packets through the network with little additional loss. This test is limited by the number of additional data streams available for use in the network. Increasing the scale of the network would provide a greater challenge for the routers and would alter the shape and magnitude of the PDR curves. In the current configuration, the most influential and limiting component is the Sun SPOT basestation connected to the sink node. Performance characteristics established by the basestation remain consistent throughout the network with the exception of slight distortion introduced by each router.

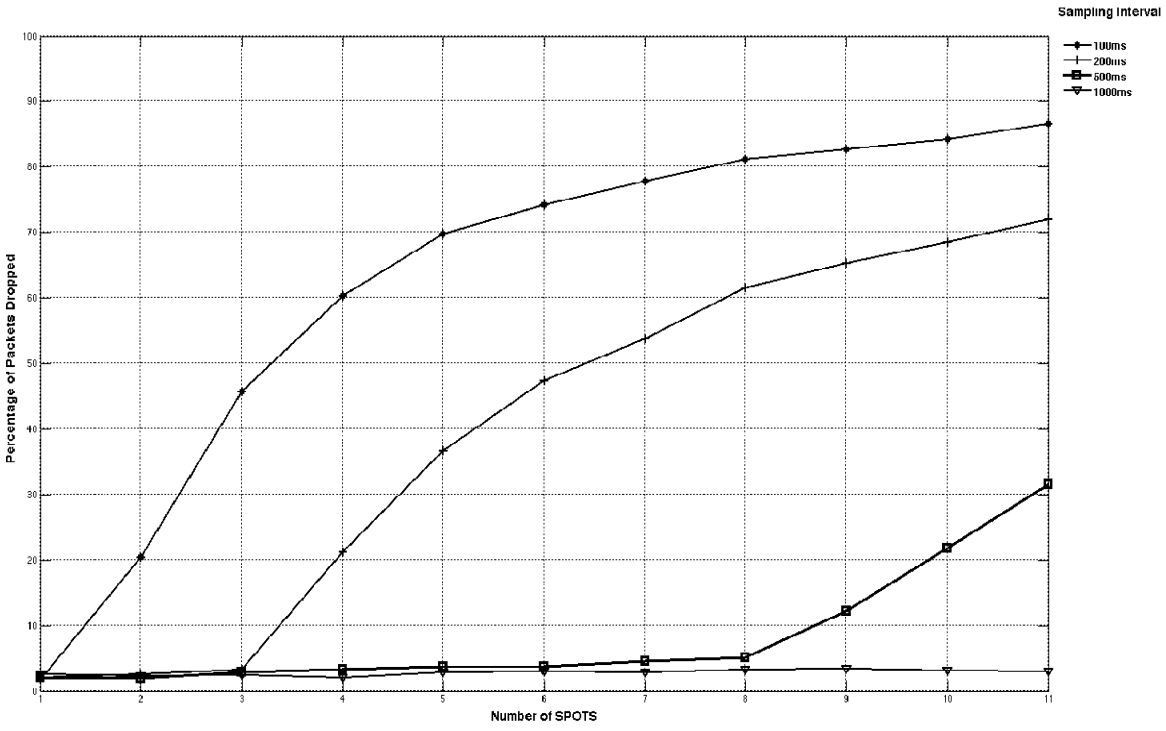

Figure 36. Packet Drop Rate Across A Two Node Wireless Relay (3 Heterogeneous Data Streams)

Applying the Adaptive Sampling Algorithm, the packet loss for networks with node densities of one through four SPOTS is examined. Figures 37 and 38 show plots of the PDR for these networks at a sampling interval of 1 packet/100 ms. As shown in the plots, the algorithm has greatly reduced the PDR compared with the original results shown in Figure 32. In the original experiment, the PDR increased from 1.4% to 20%, to 45% and finally 60% at a node density of four. With the algorithm in place, the maximum PDR is 7% and occurs at a node density of four. While throughput is sacrificed to maintain tolerances, the algorithm significantly decreases the PDR across the networks.

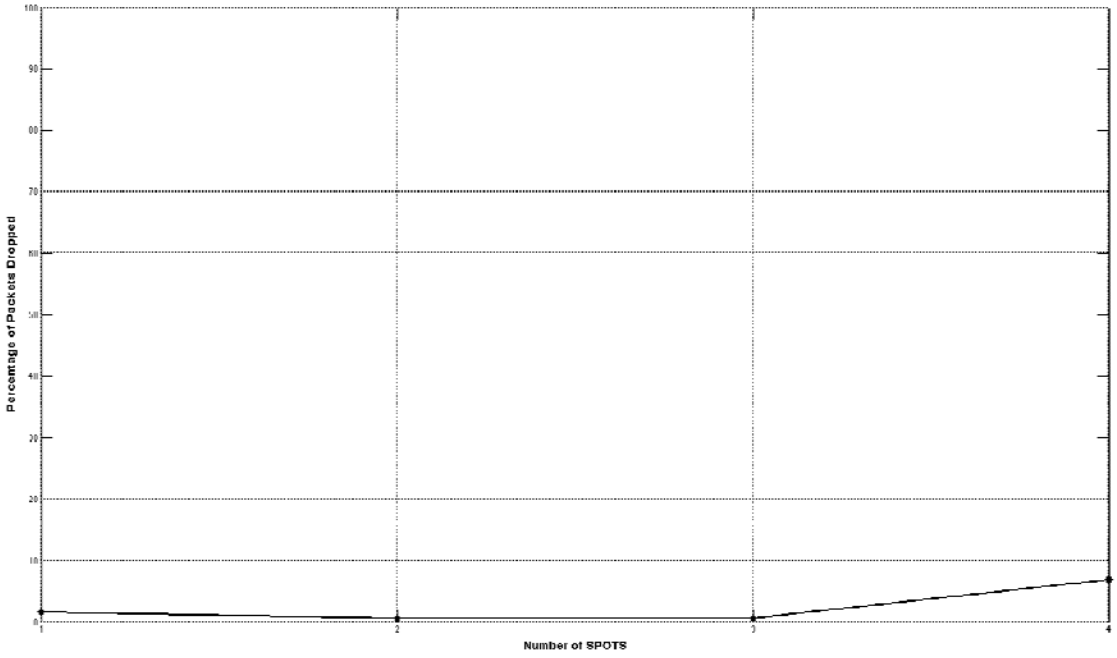

Figure 37. Sensor Field Packet Drop Rate With Adaptive Sampling Algorithm

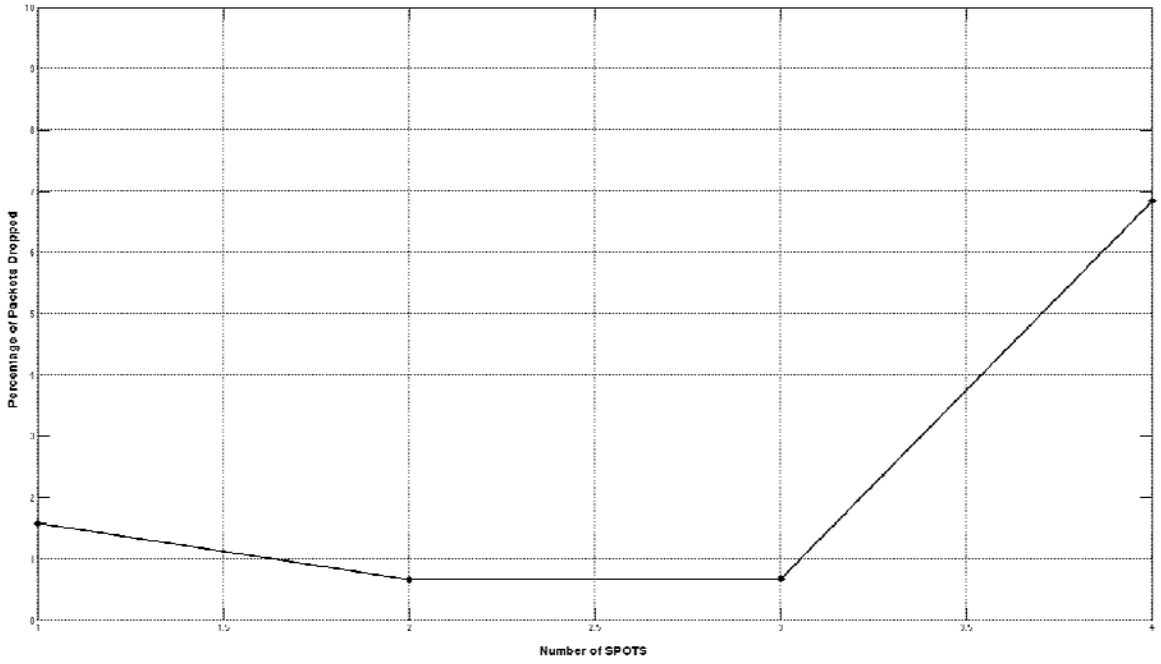

Figure 38. Sensor Field Packet Drop Rate With Adaptive Algorithm (Enlarged View)

B. PACKET LATENCY

Multi-media WSN applications, such as VOIP and streaming video, are sensitive not only to the number of packets dropped but also to the latency of these packets. Packets, which experience too much delay in a VOIP conversation, can degrade the quality of the conversation such that there is a couple of seconds delay from when one user talks into the phone until the other user hears it, similar to older radio communications.

Real time applications, such a streaming video, are sensitive to excessive delay in the network. Excessive delay can degrade the quality of the video signal enough to make it of no value. While periodically sampled signals can generally tolerate much more delay, three seconds will be used as the maximum allowable delay in the network.

Delay was tested at each stage of development of the network, i.e. sensor field/sink node, sensor field to sink node plus one wireless router, and sensor field to sink node plus two wireless routers. To test the delay, periodic data streams are transmitted

from varying numbers of nodes while the sampling rate is held constant. This was repeated for multiple sampling rates to observe the effect on end-to-end delay. To measure delay, each packet is marked with a timestamp at creation. System time is synchronized when the test application is loaded onto the SPOT. The packet time stamp is compared with the system time arrival to measure the end-to-end delay. Because we are interested in the QoS provided to the end user, packet processing time at each link in the network is incorporated into the end-to-end delay figure.

Because the load across the network during the test is not sufficient to stress the wireless routers, the delay characteristics measured at the sink remain consistent throughout the network. The only difference in the values is introduced by the variable delay across each wireless link. Figure 39 shows a plot of the delay at a sampling rate of 100 ms as measured at the sink node. As the node density increases, the amount of end-to-end delay experienced by the packets also increases. Delays in the network converge into three distinct regions based on node density and sampling rate. At 100 ms sampling rate, the delay in networks with one or two nodes converges to approximately 1000 ms. Networks with three, four or five nodes converge to the region around 11000 ms to 12000 ms. Networks with node densities greater than five converge to approximately 15000 ms.

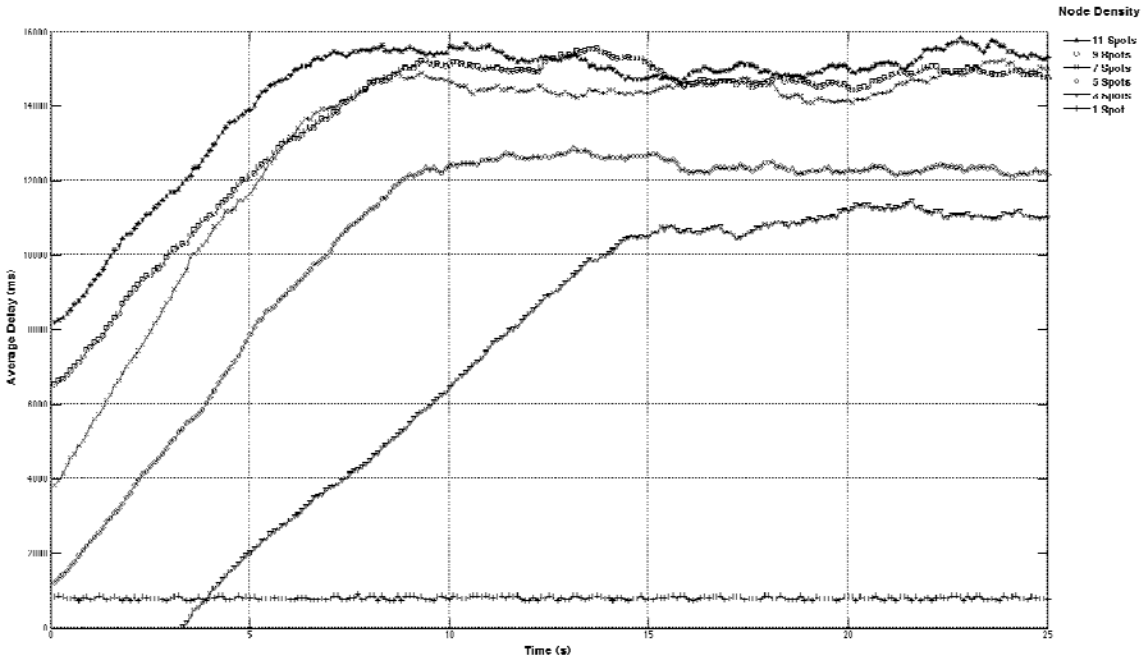

Figure 39. Sensor Field Packet Latency (Sample Interval 100 ms)

By the increasing the sampling interval, we are able to show that the end-to-end delay begins to "flatten out" as the interval gets larger. As shown in Figures 40 through 42, networks with smaller node densities begin to converge to 1000 ms, while the networks that remain in the upper region of the figures begin to converge to a smaller delay of 14000 ms. As Figure 42 shows, at a sampling interval of 1000 ms, all of the networks have converged to an end-to-end delay of 1000 ms regardless of the node density. This indicates that the traffic is not heavy, which can cause severe delays.

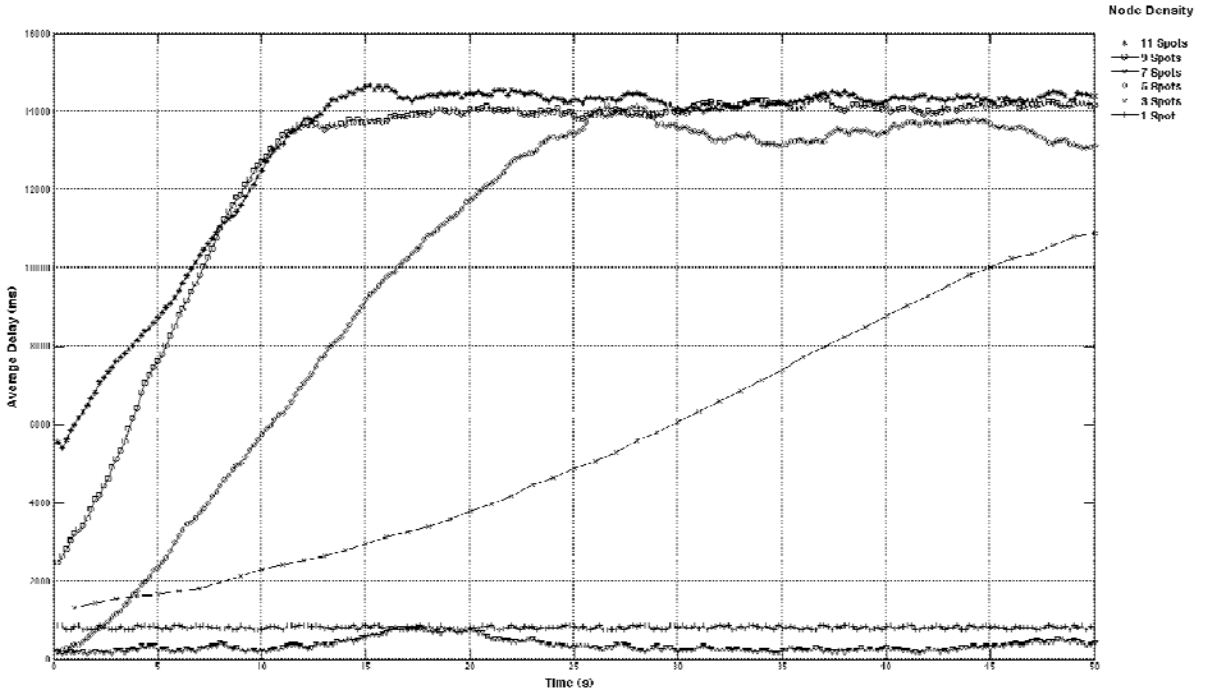

Figure 40. Sensor Field Packet Latency (Sample Interval 200 ms)

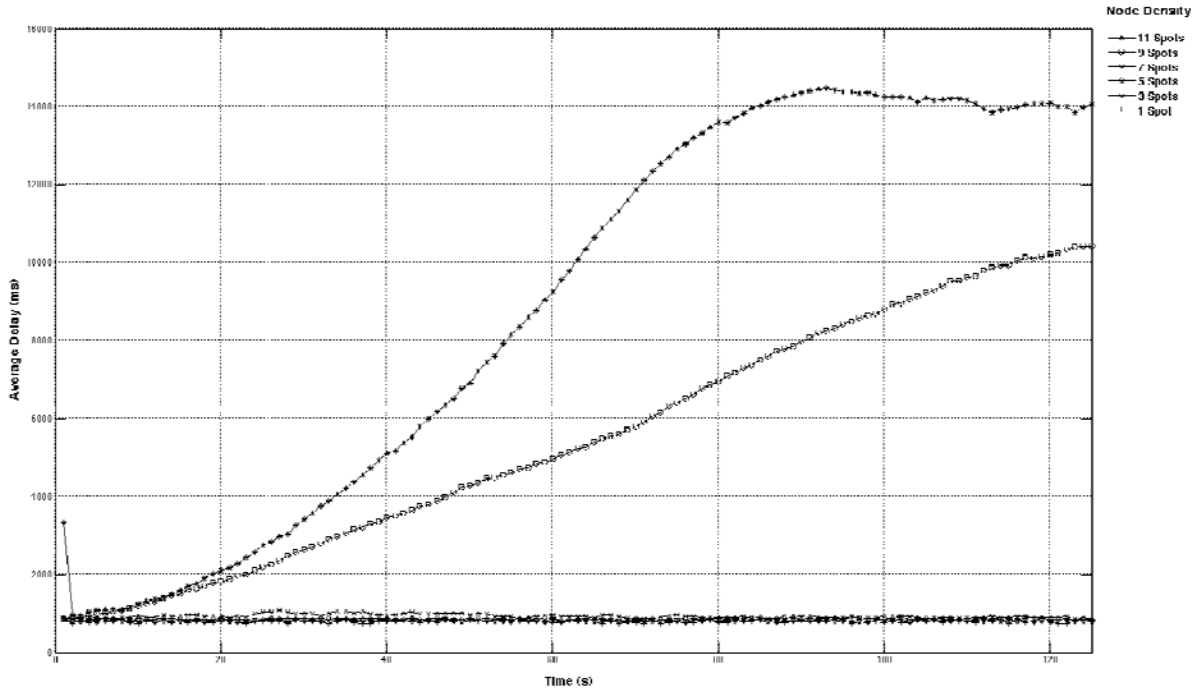

Figure 41. Sensor Field Packet Latency (Sample Interval 500 ms)

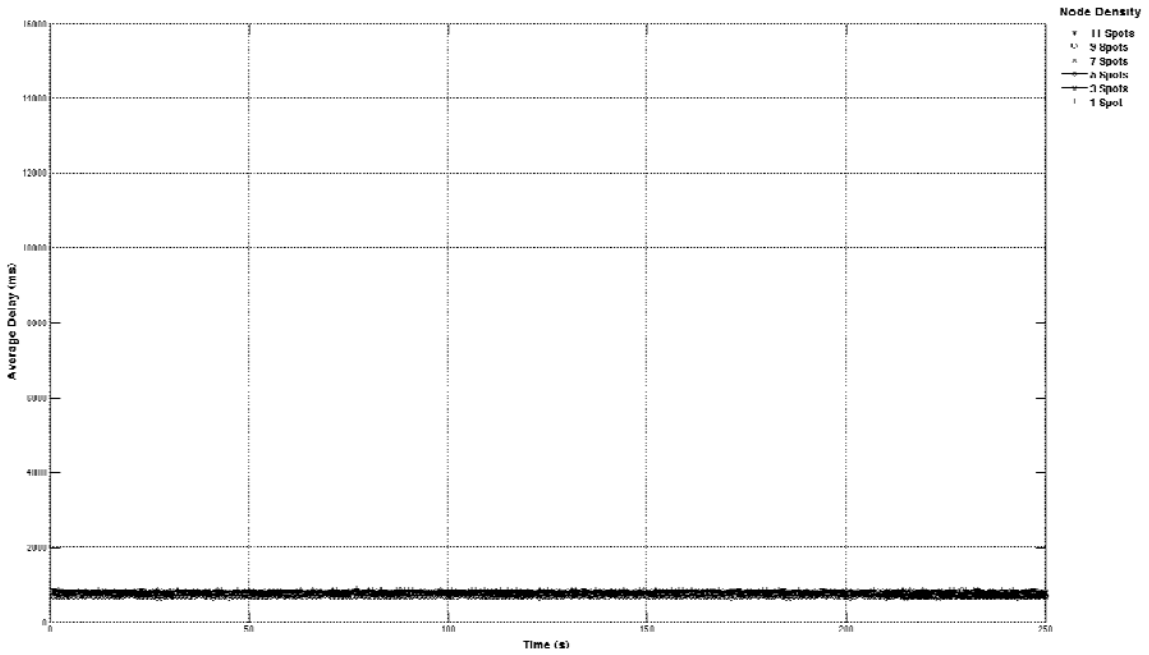

Figure 42. Sensor Field Packet Latency (Sample Interval 1000 ms)

As is assumed in the characterization of the sensor field, the basestation attached to the sink node is the component which governs the performance of the network. This assumption is reinforced when the delay across the relay links are evaluated. Figures 43 and 44 show the end-to-end delay across the network when configured with single and dual relay nodes. These plots show network delay at a sampling interval of 200 ms with varying node densities. Comparing these plots with the results from the sensor field at a sampling interval of 200 ms, Figure 40, it can be seen that the basic curves remain with a minor variation introduced with the addition of each node.

To examine the impact of the Adaptive Sampling Algorithm, it is tested on networks with node densities ranging from one to four. The initial sampling interval is set to 100 ms with an adjustment factor of 100% for each sampling adjustment. The threshold level for the algorithm is set at 2000 ms. Figure 45 shows the resulting plot of network delay. As can be seen, the network begins to exhibit the same increasing delay characteristics as in Figures 39 through 41. However, as the delay crossed the threshold

value the algorithm slows the sampling rate and returns the delay to approximately 1000 ms. Although not fully optimized, the basic Adaptive Sampling Algorithm shows the potential for maintaining network stability.

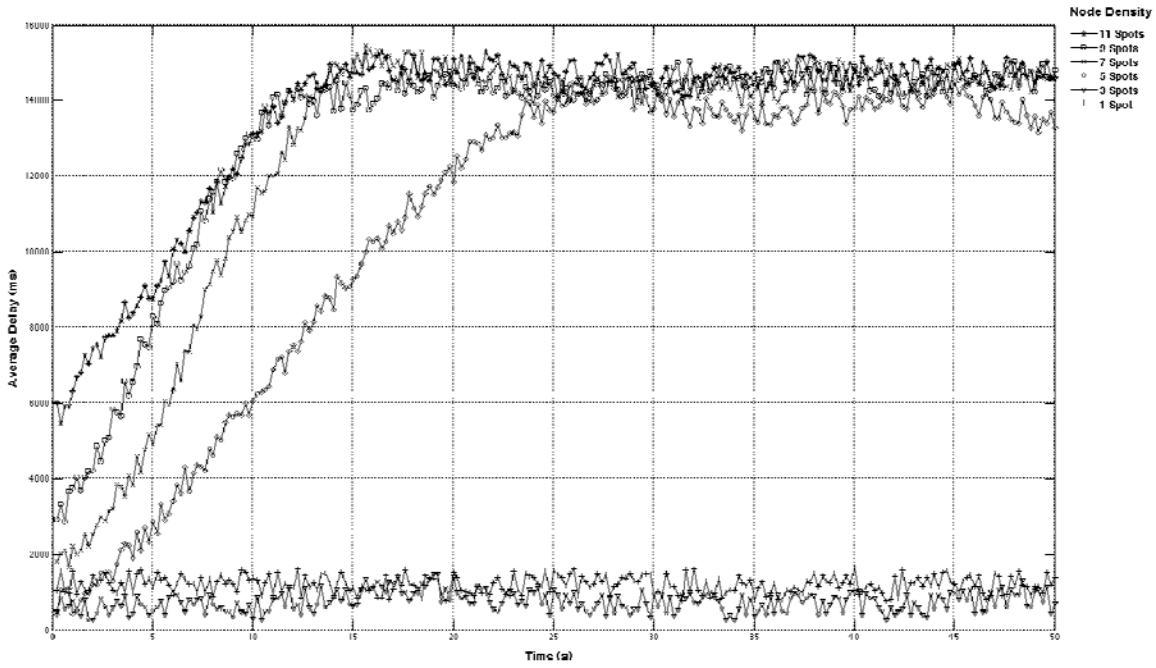

Figure 43. End-to-End Network Delay Across a Single Relay (200 ms Sample Interval)

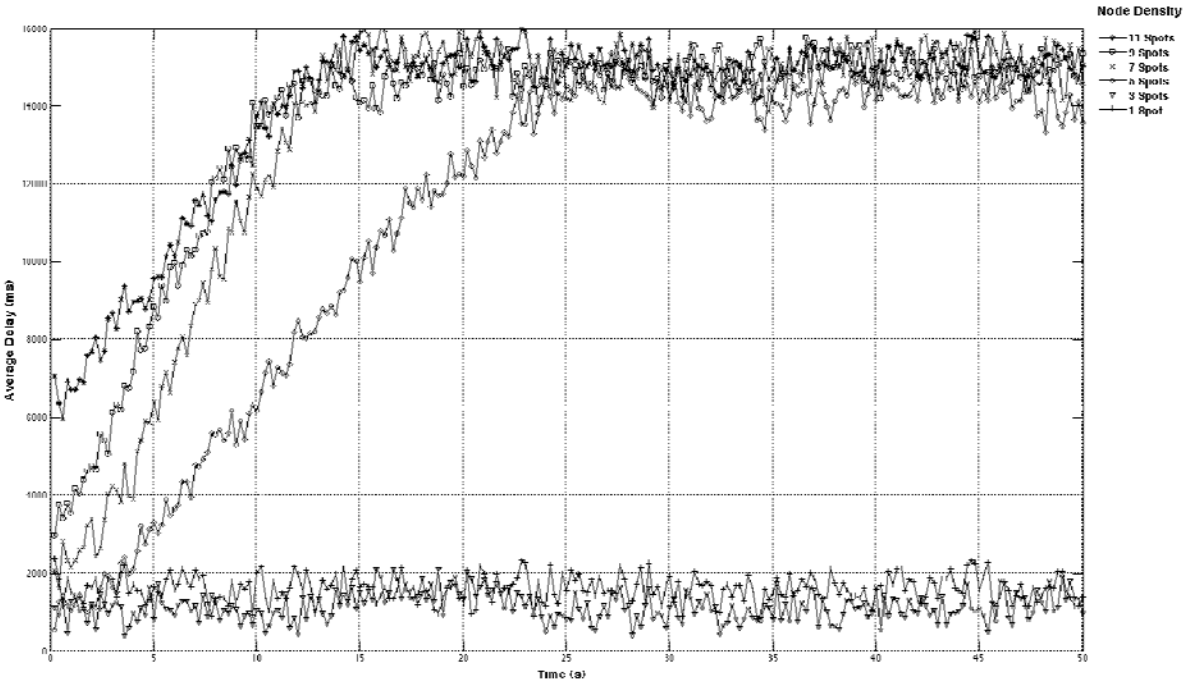

Figure 44. End-to-End Network Delay Across Two Relays (200 ms Sample Interval)

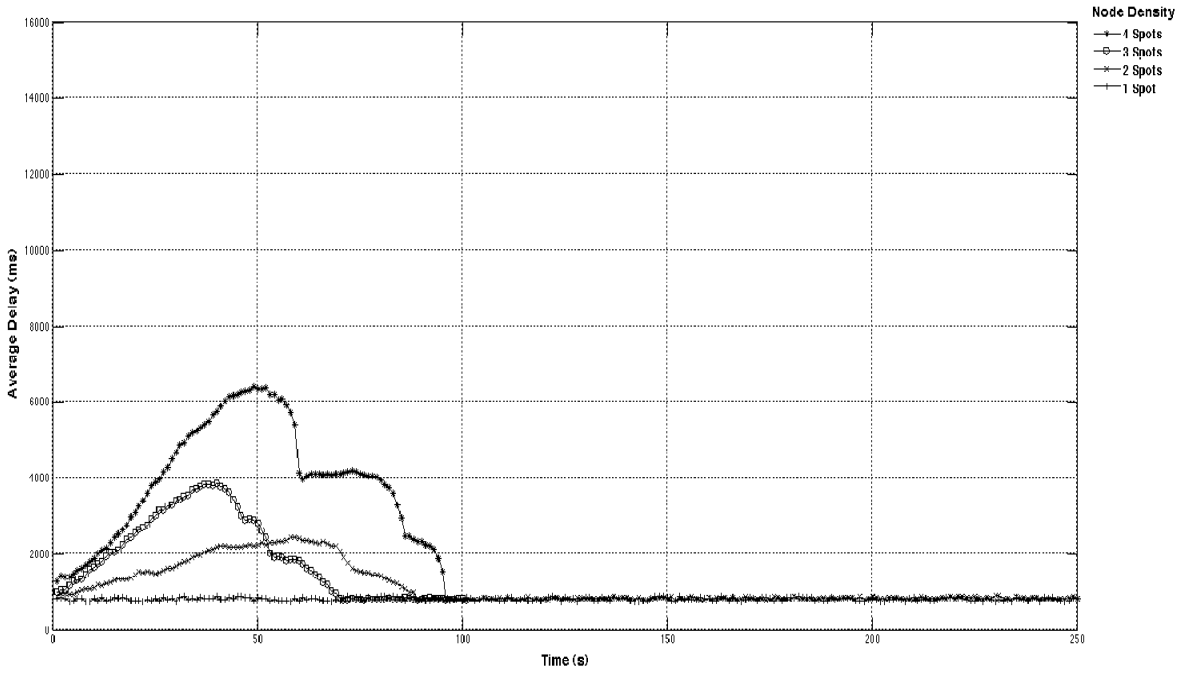

Figure 45. Network Delay With Adaptive Sampling Algorithm Implemented

C. PACKET JITTER

Applications such as streaming video and VOIP are highly susceptible to packet jitter. Since the network is being designed to support multi-media operations, jitter is an important consideration in the development of the network. For the purposes of this network, the maximum allowable jitter is set at 200 ms. Beginning with the sensor field, packet jitter from the periodic data stream is measured and characterized. Jitter is measured at varying nodes to confirm the dependence of the network on the performance of the Sun SPOT basestation. Figures 46 through 51 show the packet jitter as measured at the sink node. It is observed in Figures 47, 49 and 50 that a significant amount of jitter exists at the beginning of the plot. After repeated experimentation, it is determined that this increased jitter is the result of the initial network start-up, configuration and route discovery. After this short period of time, network jitter stabilizes until the increased traffic across the network begins to fill the buffers on the SPOTs. This causes the dramatic spike seen in Figures 46 through 50. As with the other metrics evaluated, jitter is a function of the sampling interval. As Figure 51 shows, jitter is minimized at a sampling interval of 1000 ms.

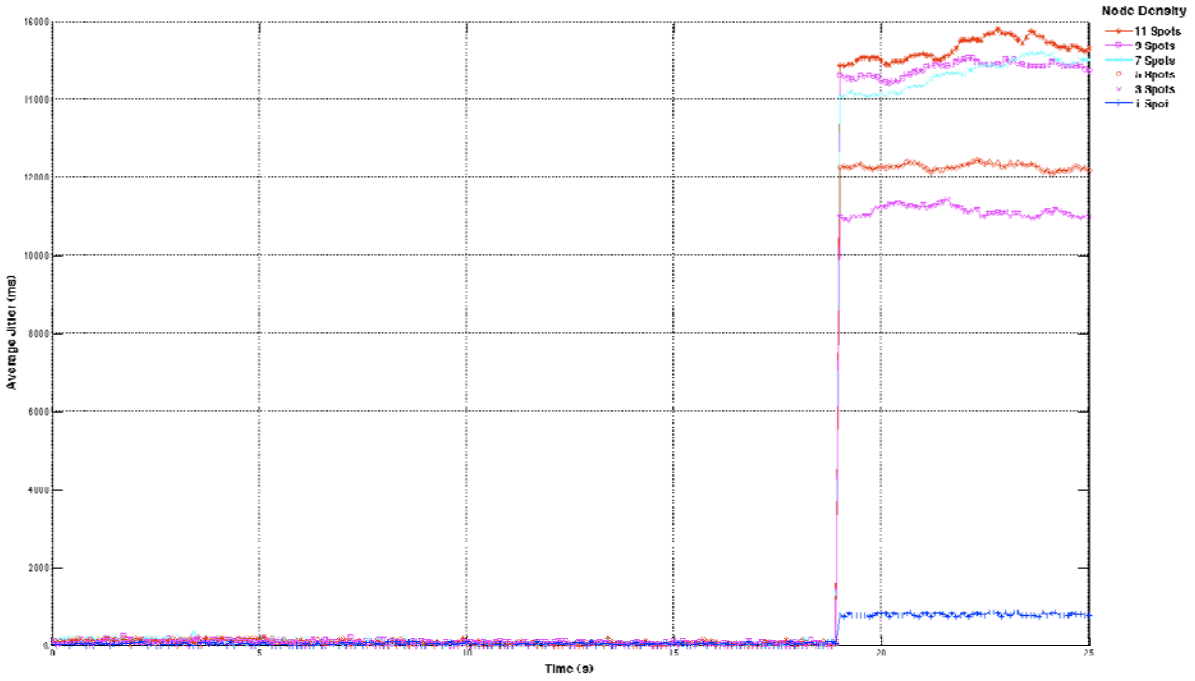

Figure 46. Sensor Field Packet Jitter (Sample Interval 100 ms)

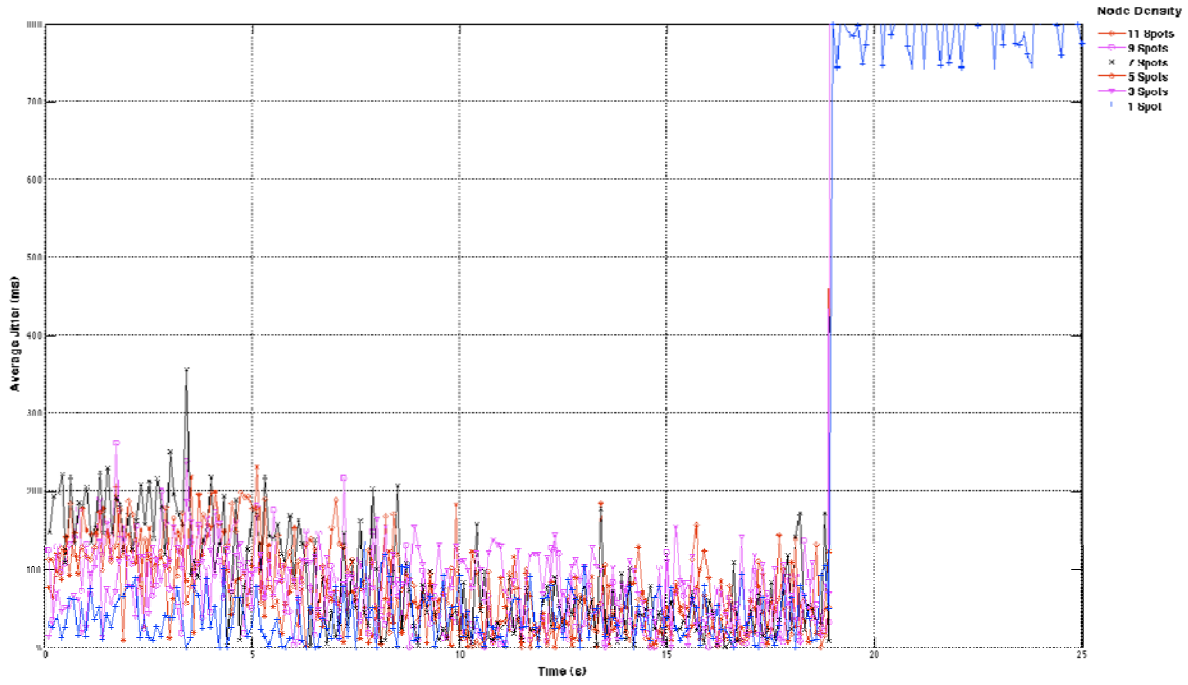

Figure 47. Sensor Field Jitter (Enlarged View – Sample Interval 100 ms)

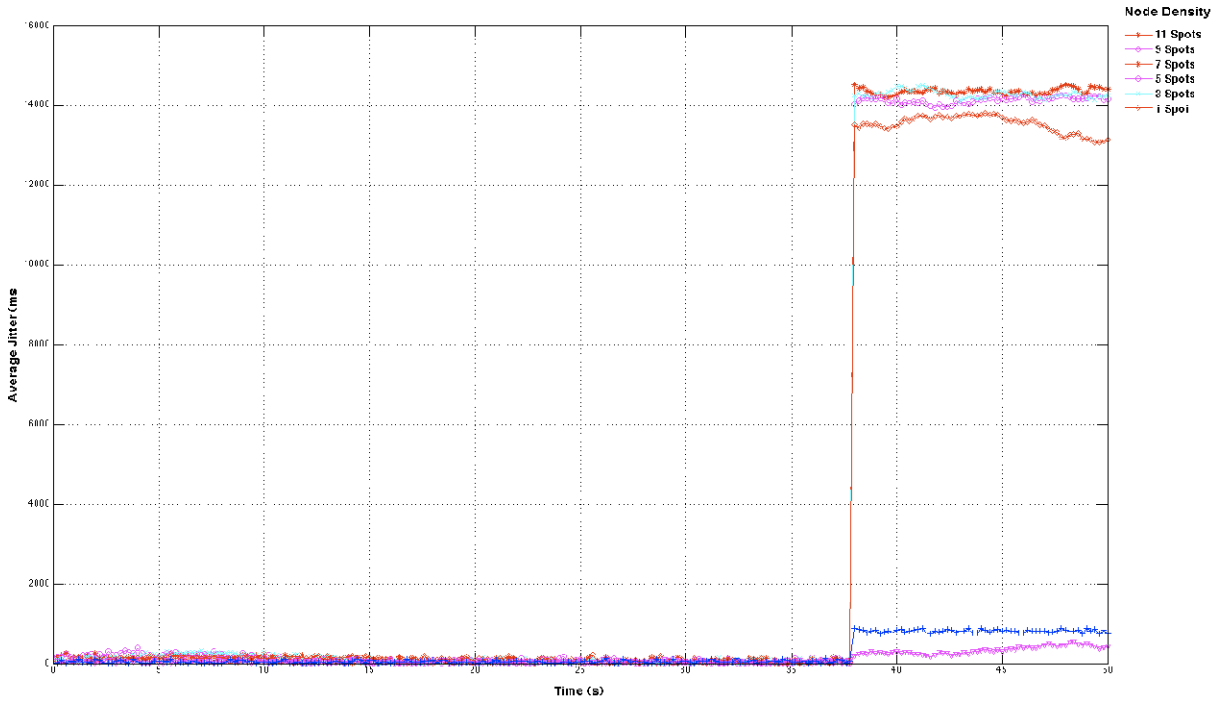

Figure 48. Sensor Field Packet Jitter (Sample Interval 200 ms)

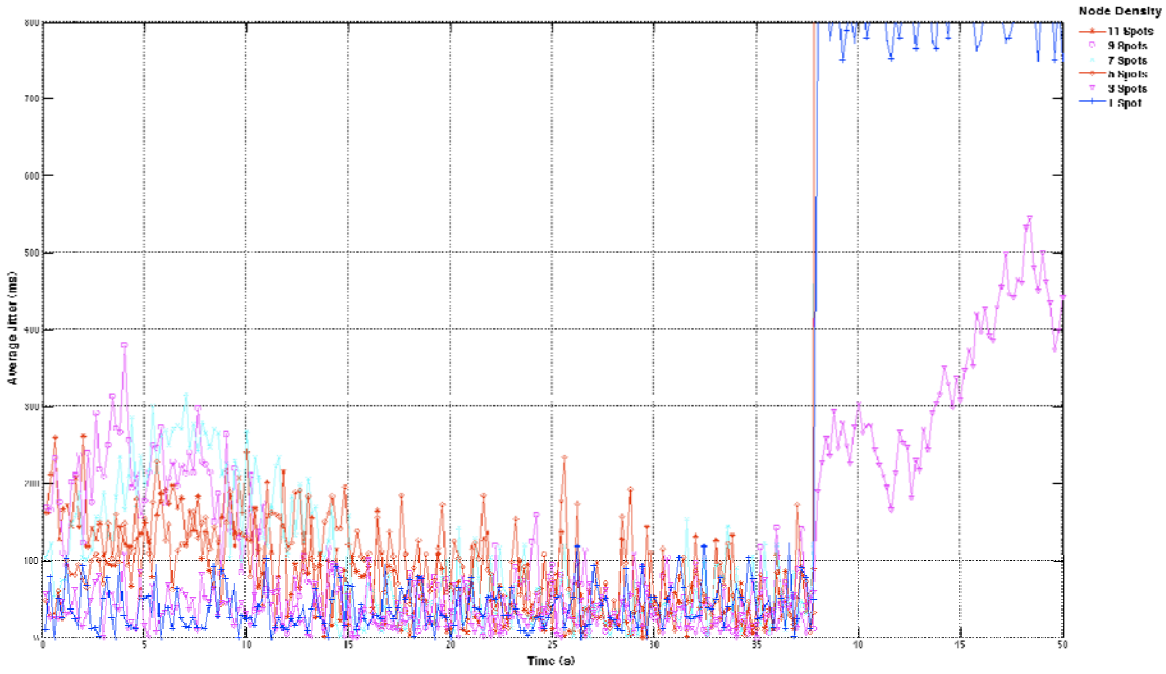

Figure 49. Sensor Field Packet Jitter (Enlarged View - Sample Interval 200 ms)

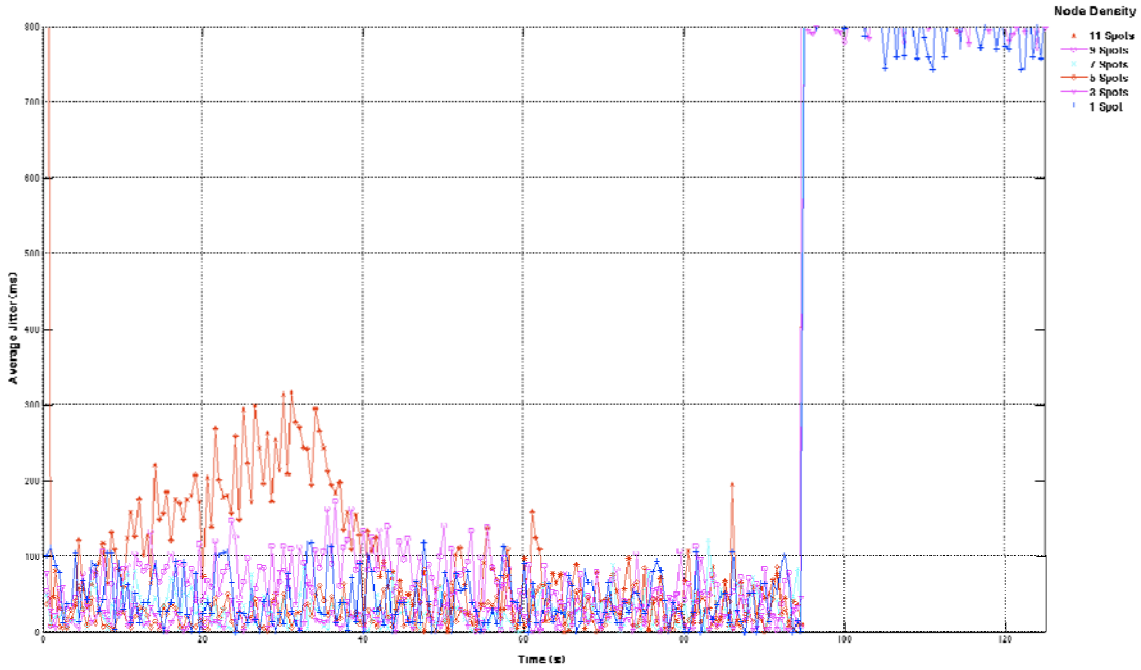

Figure 50. Sensor Field Packet Jitter (Enlarged View - Sample Interval 500 ms)

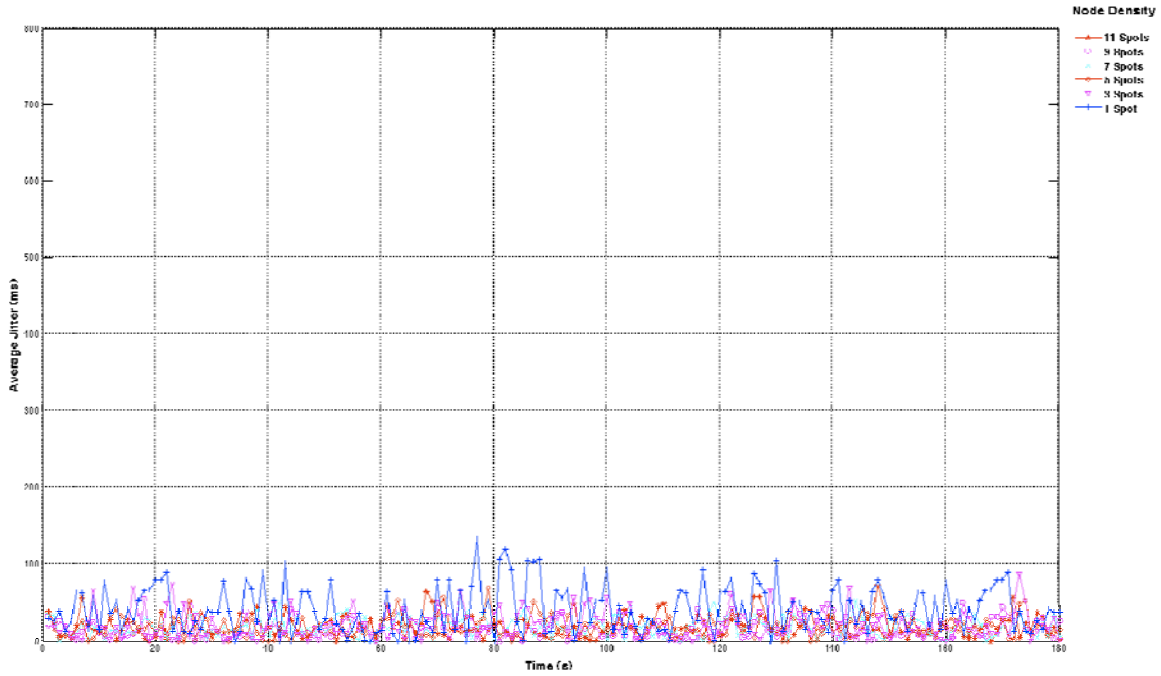

Figure 51. Sensor Field Packet Jitter (Sample Interval 1000 ms)

D. SUMMARY

This chapter examined the results of the experiments used to characterize the network. The primary performance metrics of Packet Drop Rate, Latency and Jitter were examined in reference to the network. Node density as well as sampling interval were critical design elements that effected the performance of the network. Network designers must make the tradeoff between node density and sampling rate in order to meet the Quality of Service requirements of their intended application. The chapter also examined how the Adaptive Sampling Algorithm affected the performance of the network. Initial evaluations showed that implementation of the basic algorithm improved network performance significantly.

The following chapter will conclude the thesis by presenting final conclusions about the research and findings. It also provides recommendations for future work and development of the mobile sensor network.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

A. CONCLUSIONS

This thesis provides a performance analysis of an integrated COTS WSN. Components used were evaluated for compatibility, and modifications were made as necessary to meet the requirements of the network. The performance of the network was evaluated at each stage of the integration process to study the impact of additional components on network performance. Packet drop rate, network delay and packet jitter were all considered when evaluating the performance of the network.

Because the traffic load paced on the network during integration and testing was light, end-to-end performance in the network was characterized by the sink node. Tests showed that performance was a trade off between node density and sampling rate. At a higher node density, the faster the sampling rate, the greater the percentage of packet loss. At a node density of eleven, the peak packet loss approached 90% at 10 samples/second. While this was an extreme case, packet loss above 5% could render sensitive applications such as VOIP and streaming video unusable. To achieve stability, network designers would have to trade off increased coverage of the phenomena for network stability. The characteristic trends shown by the initial performance curves maintained their shape, with only minor variations in value, with each additional component added to the network.

Delay was also a function of both node density and sampling rate. At high sampling rates the delay in the network tended to converge to three regions as the node density was increased. Networks with node densities of one or two experienced an average delay which converged to one second. Networks with node densities between three and five tended to converge to twelve seconds delay while densities above five converged to fourteen seconds of delay. At low sampling rate, such as sampling at every 1000ms, these delay curves tended to converge to one second delay regardless of the node density.

Use of an Adaptive Sampling Algorithm, which monitored delay experienced at the sink node, was able to control the flow rate of the data packets and improved performance across all metrics. Packet loss was reduced from a high of 60% ,for a four node network at a sample interval of 100 ms, to 7% for the same network. End-to-end delay was also improved significantly. As delay began to increase in the network, the algorithm was able to dynamically adjust the sampling rate and the resulting average delay converged to one second. While the Adaptive Sampling Algorithm is a basic means of flow control, the algorithm demonstrates the improved performance that can be realized by controlling network parameters at an individual node level.

This thesis demonstrated the viability of COTS components to form a mobile WSN. Using this basic network model, additional development will allow the network to meet the requirements of a military WSN.

B. FUTURE WORK

This thesis presented an initial approach to integrating commercial-off-the-shelf components into a wireless sensor network capable of supporting military operations. Considerable amount of work remains, such as increasing the scale of the network and optimizing the performance of the components and algorithms.

The maximum number of sensors used in this thesis was 11 Sun SPOTs and two Internet video cameras. WSNs deployed in real world applications could require a greater number and variety of sensors. Adding additional types of sensors, such as seismic and audio transmitting devices, would allow greater diversity in the type of network.

Power consumption is one of the most critical factors in a WSN. WSNs used in military applications are generally deployed in hostile or remote locations. This limits the amount of servicing that may be done to maintain the network. The current network does not address the issue of power consumption. Future versions of the network should explore the use of an adaptive power algorithm. Controlling the amount of power consumed by the sensor nodes would extend the life of the network.

Initial efforts to use the Nokia N800 as a sink node were unsuccessful due to compatibility issues with the operating system. As a result a laptop computer served as the sink node for the network. In order for the concept to be viable, the sink node must be small enough in size as to be concealed when deployed. The component chosen must also be JAVA compatible to allow for the developed translator and Adaptive Sampling Algorithm to be deployed.

The IEEE 802.15.4 to IEEE 802.11 translator developed during this thesis is a JAVA based software program. Future versions of the network may include a hardware translation device that could be plugged into a component's USB port. Using a hardware based solution, it would eliminate the requirement for JAVA compatibility and allow a greater range of commercial devices to be incorporated into the network.

The initial GUI provides basic functionalities for a relatively limited number and type of sensors. As the size of the network and the variety of sensors increase a more robust GUI will be required. It is recommended that future versions of the GUI can be adapted for large scale sensor networks.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] David S. Alberts, John J. Garstka, and Frederick P Stein. *Network centric warfare: Developing and leveraging information superiority (2nd Edition Revised).* Department of Defense, 2000.

[2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, " Wireless sensor networks: a survey," *Computer Networks-The International Journal of Computer and Telecommunications Networking*, Vol. 38, No. 4, pp. 393-422, March 2002.

[3] National Science Foundation, *"National patterns of R&D resources: 2006 Data Update,"* http://www.nsf.gov/statistics/nsf07331/, August 2008.

[4] Geoffrey Werner-Allen, Konrad Lorincz, Mario Ruiz, Omar Marcillo, Jeff Johnson, Johnathan Lees and Matt Welsh, "Deploying a wireless sensor network on an active volcano," *IEEE Internet Computing*, pp18-25, March-April 2006.

[5] R. Szewczyk, J. Polastre, A. Mainwaring and D. Culler, "Lessons form a sensor network expedition," *LNCS 2920*, pp 307-322, 2004.

[6] W.Y. Mak, "Ad-hoc network architecture for multimedia networks," Master's Thesis, Naval Postgraduate School, Monterey, California, 2007.

[7] Sunspotworld.com, http://www.sunspotworld.com/forums/viewtopic.php?p=2862 - 2862, September 2008.

[8] G.L. Pore, "A performance analysis of routing protocols for adhoc networks," Master's Thesis, Naval Postgraduate School, Monterey, California, 2006.

[9] Hao Wang; Changcheng Huang; James Yan, "Efficient multiple-link adaptive bandwidth provisioning for end-to-end quality of service guarantee," *Electrical and Computer Engineering, 2006. CCECE '06. Canadian Conference on* , vol., no., pp.127-131, May 2006.

[10] Hao Wang; Changcheng Huang; Yan, J., "A feedback control model for multiple-link adaptive bandwidth provisioning systems," *Communications, 2006. ICC '06. IEEE International Conference on* , vol.3, no., pp.987-993, June 2006.

[11] PDAdb.net, "*Nokia N800 internet tablet (Nokia Gagarin) specs,* " http://www.pdadb.net/index.php?m=specs&id=1094, November 2008.

[12] PDAdb.net, " *Texas Instruments OMAP 2420 specs,* " http://www.pdadb.net/index.php?m=cpu&id=a2420, November 2008.

[13] Broadcom, *"BCM5352EL AirForce 802.11g router system-on-chip with broadRange technology,"* http://www.broadcom.com/products/Wireless-LAN/802.11-Wireless-LAN-Solutions/BCM5352EL, September 2008.

[14] Paul Asadoorin and Larry Pesce. *Linksys WRT54G ultimate hacking*. Syngress Publishing, Burlington, MA, 2007.

[15] Cisco Systems, "*Wireless Lan (WLAN) mutlipath and diversity,"* http://www.cisco.com/en/US/tech/tk722/tk809/technologies_tech_note09186a008019f646.shtml, September 2008.

[16] Linksys, *"WRT54GL downloads,"* http://www.linksys.com/servlet/Satellite?c=L_CASupport_C2&childpagename=US%2FLayout&cid=1166859841350&pagename=Linksys%2FCommon%2FVisitorWrapper&displaypage=nodata, September 2008.

[17] Open-WRT.Org, *"Open WRT Kamikazee version 7.07 download,"* http://openwrt.org/, September 2008.

[18] DD-WRT.Com, *"DD-WRT downloads,"* http://www.dd-wrt.com/dd-wrtv3/index.php , September 2008.

[19] Sun Microsystems, "*JAVA version 6 update 10 download*," http://www.java.com/en/download/index.jsp, November 2008.

[20] Apache.org, "*Apache ANT version 1.7 download,*" http://ant.apache.org/bindownload.cgi, November 2008.

[21] Sunspotworld.com, "*Project Sun Spot – Information and resources,"* https://www.sunspotworld.com/docs, November 2008.

[22] Sunspotworld.com, "*Sun Small Programmable Object Technology (Sun SPOT) developer's guide, blue release version 4.0,*" https://www.sunspotworld.com/docs/Blue/spot-developers-guide.pdf, November 2008.

[23] Texas Instruments, "*CC2420 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF transceiver,"* http://www-mtl.mit.edu/Courses/6.111/labkit/datasheets/CC2420.pdf , September 2008.

[24] Internet Engineering Task Force, "*RFC 3561 Ad hoc On-Demand Distance Vector (AODV) Routing,*" http://www.rfc-editor.org/cgi-bin/rfcdoctype.pl?loc=RFC&letsgo=3561&type=http&file_format=txt, September 2008.

[25] Roger Meike's Blog, "*Project Sun Spot,"* http://blogs.sun.com/roger, November 2008.

[26] David Simmons Blog, "*On-the Spot*," http://blogs.sun.com/davidgs/, November 2008.

[27] Sunspotworld.com, "*Sun Spots discussion*," https://www.sunspotworld.com/forums, November 2008.

[28] Netbeans.org, "*Netbeans IDE version 6.5 download*," http://www.netbeans.org/, November 2008.

[29] Dlink.com, "*DCS-5220 pan/tilt internet camera*," http://www.dlink.com/products/?pid=546, November 2008.

[30] DD-WRT.com, "*Installation wiki,"* http://www.dd-wrt.com/wiki/index.php/Installation, November 2008.

[31] DD-WRT.com, "*Wireless bridge - DD-WRT wiki,"* http://www.dd-wrt.com/wiki/index.php/Wireless_Bridge, November 2008.

[32] DD-WRT.com, " *Repeater bridge - DD-WRT wiki*," http://www.dd-wrt.com/wiki/index.php/Repeater_Bridge , November 2008.

[33] Sunspotworld.com, *" Sun Spot forum - operating temp, sampling vs transmitting rate, rang*e," https://www.sunspotworld.com/forums, November 2008.

[34] ITU-T Recommendation P.59, *"Artificial conversational speech,"* http://www.itu.int/rec/T-REC-P.59/en , September 2008.

[35] Li Ji, Xia Yin, Xingang Shi and Zhiliang Wang, "Conversational model based VOIP traffic generation," *Third International Conference on Networking and Services*, 2007.

[36] ITU-T, *"Recommendation G.711 pulse code modulation (PCM of voice frequencies,"* http://www.itu.int/rec/T-REC-G.711/en, October 2008.

[37] IEEE, "*Part 15.4: Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs),"* http://ieeexplore.ieee.org.libproxy.nps.edu/stamp/stamp.jsp?arnumber=4299496&isnumber=4299495, November 2008.

.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Chairman Code EC
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California

4. Professor Weilian Su
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California

5. Professor John C. McEachen
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California

6. Professor Murali Tummala
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California